MODELING AND PREDICTION OF THERMALLY INDUCED ERRORS IN MACHINE TOOLS USING A LASER BALL BAR AND A NEURAL NETWORK

BY

NARAYAN SRINIVASA

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

1994

To my beloved parents and my wife, Anitha

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my advisor, Dr. John C. Ziegert, for his guidance and support throughout this research. His insight, technical assistance and continuous encouragement were of invaluable assistance in the completion of this research. Thanks are due to Dr. Scott Smith for his interest and helpful suggestions. I am grateful to Dr. Jiri Tlusty for his willingness to serve on my committee. It was definitely an honor to have a pioneer in machine tool calibration review this research. I am also thankful to him for allowing me to use the facilities of the Machine Tool Research Center. Thanks are also due to Dr. Carl Crane and Dr. Baba Vemuri for their useful suggestions and willingness to serve on my committee. I would also like to thank Dr. Placid Ferriera of the University of Illinois at Urbana-Champaign for allowing me to use the CMM for this research.

I would also like to thank the following members of the Machine Tool Research Center:

* Mr. Chris Mize for his valuable assistance throughout this research. His excellent design skills were an important reason for the rapid development of the Laser Ball-Bar without which this research would have been terribly set back.
* Mr. Daniel Sumanth for the development of the PC-based error compensation

system.

* Mr. William Winfough for his assistance in programming the data acquisition system for the laser.
* Mr. David Smith, Mr. Yong-Hoon Chang and Mr. Yue-Che Chen for their assistance in matters related to electronic hardware.
* Mr. Devendra Singh Rao for his assistance in setting up the capacitance gage during spindle drift measurements.
* Mr. Howard Purdy for his assistance in machining all the fixtures and the magnetic sockets.
* All members of the Machine Tool Research Center for their advice and friendship.

Finally, I would like to thank my parents for their constant encouragement and my lovely wife, Anitha, for her love and affection, without which this research would not have been possible.

TABLE OF CONTENTS

ACKNOWLEDGMENTS iii

ABSTRACT vii

CHAPTERS

1. INTRODUCTION 1

2. CHARACTERISTICS OF MACHINE TOOL ERRORS . . 5

Spatially Variant Characteristics 6
Thermally Variant Characteristics 8

3. LITERATURE REVIEW 11

Motivation 21
Research Objective 23
Scope of Research 27

4. ARTIFICIAL NEURAL NETWORKS 31

An Overview of Neural Networks 31
Working Principle of Fuzzy ARTMAP Neural Network 36

5. THE LASER BALL BAR 43

LBB Based Total Positional Error Measurement Technique . 47
LBB Based Measurement of Spindle Thermal Drifts 56

6. EXPERIMENTAL RESULTS 65

Tool Eye Setting for the LBB 65
Measurement of Rotation Matrix 68
Location of Thermocouples 69
Results 72

7. THE ERROR COMPENSATION SYSTEM 97

Working Principle of the ECS System 98
ECS Hardware 99
ECS Software 103

8. PERFORMANCE EVALUATION 107

Shape of Test Part and Cutting Conditions 107
Cutting Procedure 109
Measurement Results 110
Error in Diameters 111
Error in Lengths 112
Error in Taper 112
Error in Hemisphere Diameters 114

9. CONCLUSIONS AND FUTURE WORK 117

Conclusions 117
Future Work 118

APPENDICES

A. FUZZY ARTMAP NEURAL NETWORK IMPLEMENTATION IN MICROSOFT C 120

B. METHOD TO TRANSFORM BALL BAR COORDINATES INTO MACHINE COORDINATES . 144

C. EIA PART PROGRAM FOR TEST WORKPIECE 146

REFERENCES 149

BIOGRAPHICAL SKETCH 157

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of
the Requirements for the Degree of Doctor of Philosophy

MODELING AND PREDICTION OF THERMALLY INDUCED ERRORS IN MACHINE TOOLS USING A LASER BALL BAR AND A NEURAL NETWORK

By

Narayan Srinivasa

December 1994

Chairperson: Dr. John C. Ziegert
Major Department: Mechanical Engineering

Thermally induced errors are major contributors to the overall accuracy of machine tools. While this area is well researched, a major drawback of currently proposed methods is the amount of time consumed in building this model. Due to this drawback, there has been a reluctance in the industrial community to adopt these techniques. In this study, a direct method of machine tool calibration is adopted. Rather than individually measuring the parametric errors of the machine tool over a thermal duty cycle to build the error model of the machine, the total positional errors at the cutting tool are rapidly measured using the laser ball bar. The measured errors are correlated with temperature gradients of the various heat sources in an on-

line fashion by a neural network using pattern recognition principles. The methodology is implemented on a two axis computerized numerical control turning center. The neural network builds the error model of the machine tool using training data collected over a thermal duty cycle.

The predicting ability of the network was further tested using random thermal duty cycles. A PC based system was used to compensate for the errors predicted by the neural network. The performance of the system was also evaluated through cutting tests under various thermal conditions. A substantial improvement in the overall accuracy of the turning center was obtained.

CHAPTER 1
INTRODUCTION

Machine tools are serial-link, open-loop, kinematic chains. At one end of the chain is the part to be machined and at the other is the cutting tool. One of the major functional requirements of this class of mechanisms is the ability to position the cutting tool with absolute accuracy on the order of 0.001% of the total working volume. Improvement of machine tool accuracy is an essential part of quality control in manufacturing processes. There is a constant pressure on industrial manufacturers to produce high quality products while maintaining high productivity. A survey conducted by Bowen and Duncan [1] on major industrial manufacturers shows that 90% of the cost for insuring quality is due to scrapping/reworking of products that do not lie within design tolerances. This approach is not desirable by industry because of its high cost. Therefore, the emphasis has shifted towards developing new methods that can help produce the product correctly the first time it is manufactured. These methods must be capable of providing the correct quality control actions to the machine tool by actively monitoring and correcting the error sources of manufacturing processes rather than passively inspecting machined parts.

The sources of machine errors can be divided into three main types: geometric, elastic and thermal errors. The geometric errors arise from imperfect construction and alignment of the slides and joints of the machine as well as machine

wear. These errors contribute to workpiece errors by more than 50μm in medium and large-size machine tools (i.e., with axis travel of more than 300mm) as reported by Hemingray [2]. Elastic errors result from non-rigid body effects in the kinematic elements of the positioning chain. They are quasistatic, due primarily to gravitational and or/clamping effects in the workpiece and moving elements of the machine, or dynamic (vibrational), resulting from cutting forces and inertial effects. Thermal errors arise due to thermal deformations of the machine elements caused by an extremely complex temperature field. Heat sources that exist within the structure include the leadscrew bearings and nuts, axis drive motors, spindle, friction in the guideways and heat generated by the cutting process. These heat sources are time-variant in nature and are generally thought to be the largest contributors to overall machine inaccuracy. The contribution of the three types of errors discussed thus far is presented in the form of a pie chart as shown in Figure 1-1.

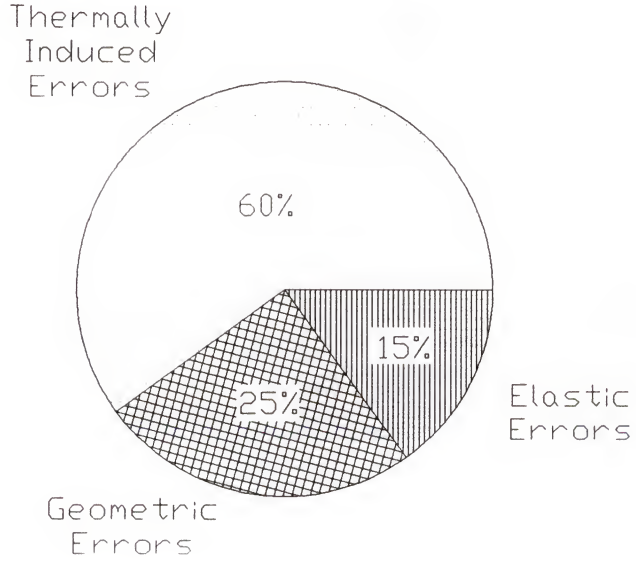


Figure 1-1 Contribution of machine tool error sources.

If these error sources can be modelled adequately and the accumulated error predicted from the model, then it should be possible to compensate for this predicted error in the machine controller by modifying the positioning commands in an appropriate manner. This type of error compensation is, in fact, the approach used in many current generation machine tools and CMMs. However, when reduced to practice, this method is costly, time-consuming and not robust to variations in machining conditions. The last is especially important in the context of unmanned manufacturing because the removal of the machinist from the machining process has created problems in maintaining machining accuracy. Experienced machinists can accommodate changes in environmental and operational conditions during manual operation of machines. This is usually the case for small batch jobs where an extensive error compensation system may not be economically feasible. In the case of medium and large batch jobs, the computer numerical controlled (CNC) machine is needed to manufacture parts. However, the CNC controller does not possess the intelligence of an experienced machinist to adjust to variations in machining conditions.

This research will be devoted to obtaining a timely, robust and cost-effective method to compensate for geometric and thermal errors in machine tools. It makes use of the self-learning properties of artificial neural networks (ANN) to predict the net positioning error at an arbitrary point in the workspace from knowledge of the error at some specified points in the workspace. This knowledge is obtained from the measurement of geometric errors and their thermal variations at the cutting tool

as well as correlation with other process variables.

The dissertation is outlined as follows. In chapter 2, the basic concepts of geometric and thermally induced errors in machine tools are outlined and the characteristics of these errors are also described. Literature related to this research is reviewed in chapter 3. This chapter also describes the motivation behind this research and outline the objectives and scope of this research. Chapter 4 describes the fundamental principles of neural networks and outlines the working principle of the neural network to be used in this research. Machine tool errors are measured using a new calibration instrument called the Laser Ball Bar(LBB). The working principle of the LBB is described in chapter 5, which also outlines the trilateration procedure used to extract thermally induced spindle errors and the total error at the cutting tool. In chapter 6, the experimental setup to measure the total positional errors at the cutting tool and thermally induced spindle errors using the LBB is described. The experimental results collected as well as the performance of the neural network for thermal error prediction are also presented. The hardware and software details pertaining to the error compensation system are delineated in chapter 7 which also outlines the approach to integrating the neural network with the error compensation system. In Chapter 8, the performance of the system is evaluated by cutting some parts on the turning center. Chapter 9 provides conclusions and suggestions for future work.

CHAPTER 2
CHARACTERISTICS OF MACHINE TOOL ERRORS

The dimensional accuracy of the workpiece to be machined is affected by the positioning errors of the various elements of the machine tool that contribute to the positioning accuracy of the cutting tool. Each machine element has six degrees of freedom and therefore six error components. These six components consist of three translations (dx, dy and dz) along, and three rotations (roll, pitch and yaw) about, the three coordinate axes as shown in Figure 2-1. These are referred to as parametric errors. In general, the parametric errors for each axis of motion are combined using spatial kinematics (usually 4 X 4 homogeneous transformation matrices) to formulate

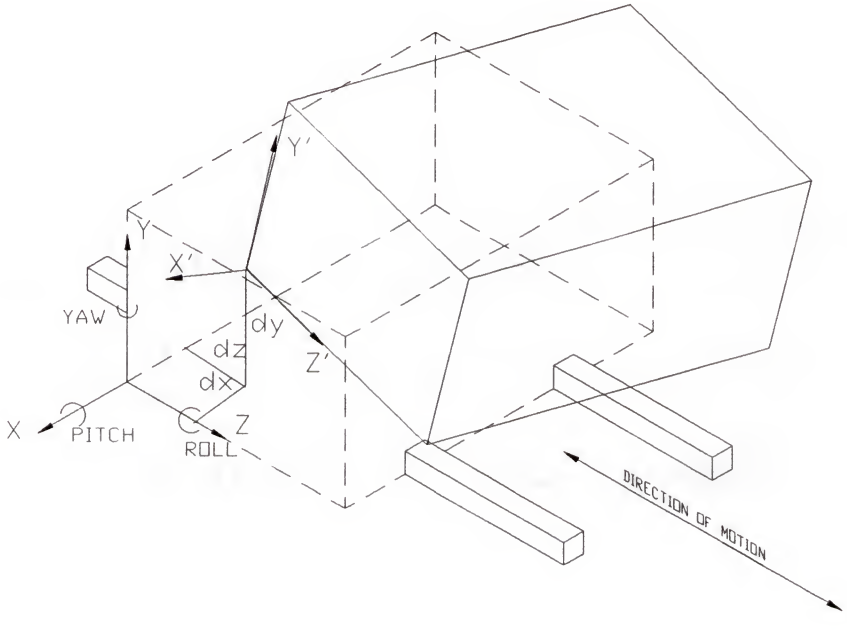


Figure 2-1 Six components of error motion of a machine component.

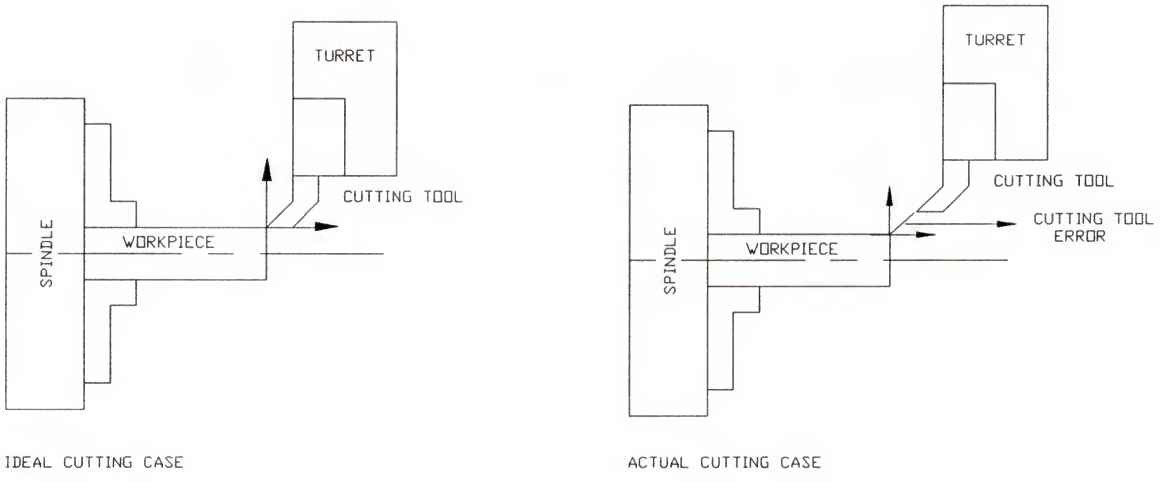


Figure 2-2 Relative error of movement between the cutting tool and workpiece.

an expression for the error of relative movement between the cutting tool and the workpiece as shown in Figure 2-2. The parametric errors have special characteristics that will now be described.

Spatially Variant Characteristics

A typical machine tool consists of slides that are stacked on each other. The motion of each slide is controlled by a servo-driven feedback loop, as shown in Figure 2-3. This loop consists of a slide, a leadscrew, a gear train, a motor and a a position encoder that acts as the feedback device. The slide is driven by the leadscrew. The leadscrew is driven by a gear train, which is in turn driven by the motor. The position of the slide is measured and fed back to the machine controller

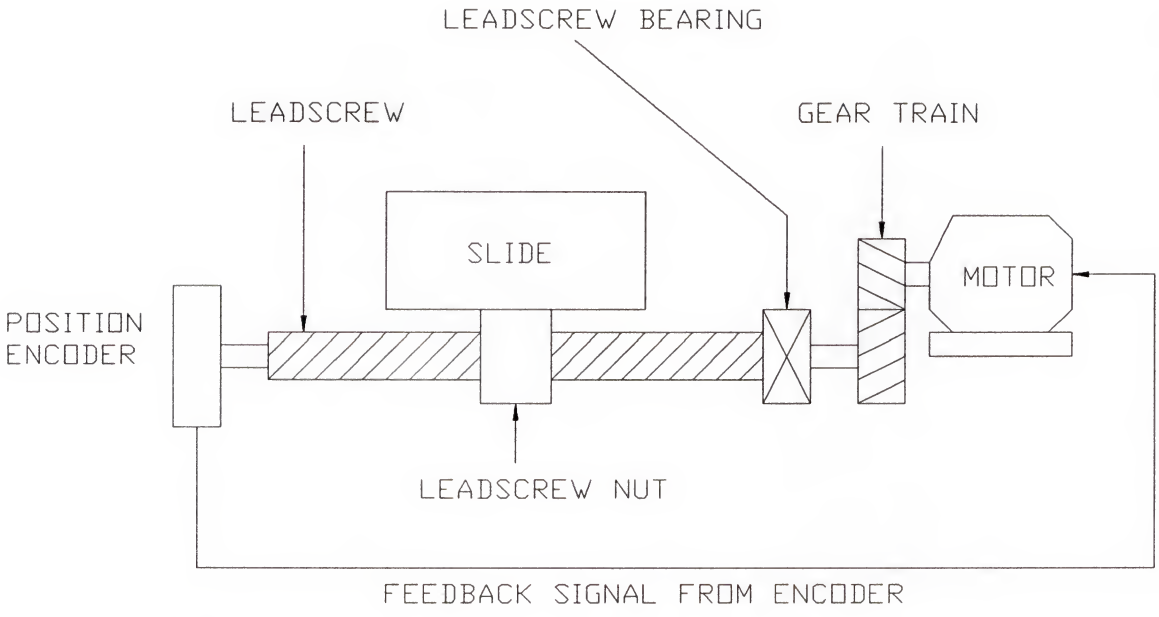


Figure 2-3 A typical servo loop of a machine tool slide.

by the encoder. Based on the position command provided by the user, an appropriate control signal (typically a voltage signal) is sent to the motor in order to position the slide to the commanded position. Due to periodicity errors and wear of the leadscrew and the offset between the position sensing device (encoder) and the actual cutting location on the workpiece (also known as the Abbe error), the machine errors vary widely over the machine workspace. In a case study, the scale and yaw errors of the z-axis of a two axis turning center were measured along different z positions using a laser interferometer system. The results of this study are shown in Figures 2-4 and 2-5. These plots show that the scale and yaw errors are spatially variant with respect to the z axis. In some machining centers, the errors due to movement along an axis may depend on the position of the other axes of motion as well. These errors are called nonrigid body kinematic errors. These errors were

found to affect machine inaccuracy for the three axis machining center used by Chen et al. [3] because of the large dead weight of the working table. However, the two axis turning center to be used in this research does not exhibit this behavior.

Thermally Variant Characteristics

In addition to being spatially variant, machine tool error sources, such as thermal errors, are thermally variant. This is due to the temperature variations in machine components caused by internal heat sources such as bearings and motors as well as temperature variation of the environment. In a case study on a turning center, the scale and yaw errors of the z-axis are measured at different thermal states using a laser interferometer system. Under cold-start conditions, the scale errors ranged from 0 to 10μm while the yaw errors ranged from 0 to 8 arcsecs. However, these errors increased by more than 70μm and 30 arcsecs within six hours of warming, as shown in Figure 2-6 and 2-7. The warming was done by exercising the carriage along the entire length of travel of the z axis.

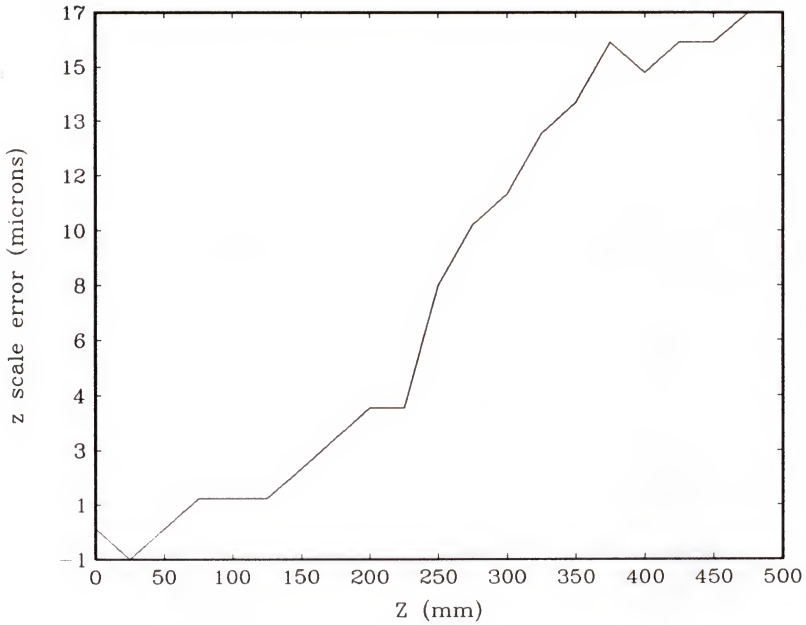


Figure 2-4 Spatially variant z scale error of a two axis turning center.

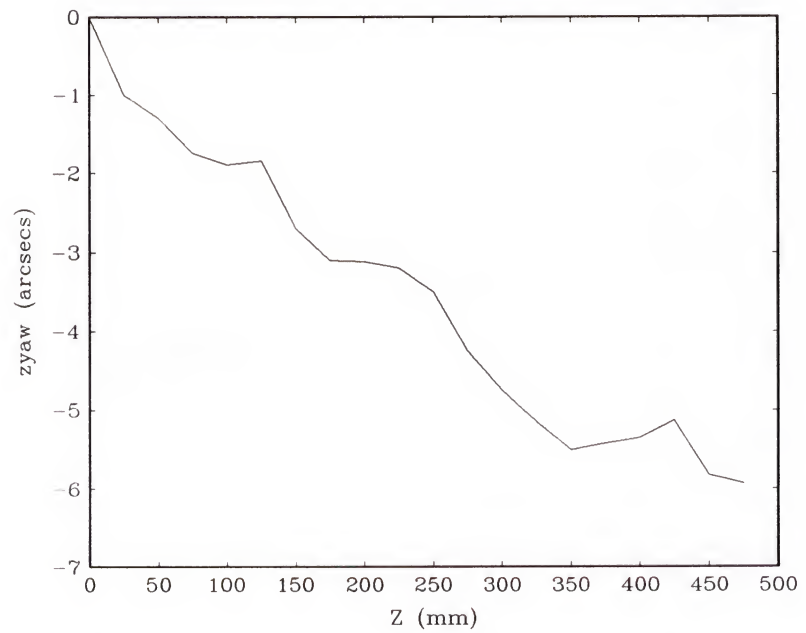


Figure 2-5 Spatially variant z yaw error of a two axis turning center.

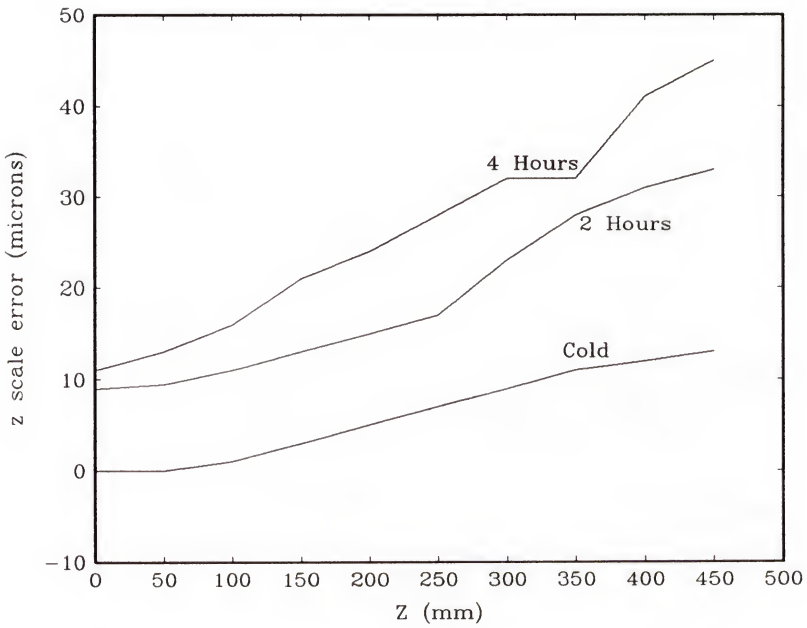


Figure 2-6 Thermally variant z scale error of a two axis turning center.

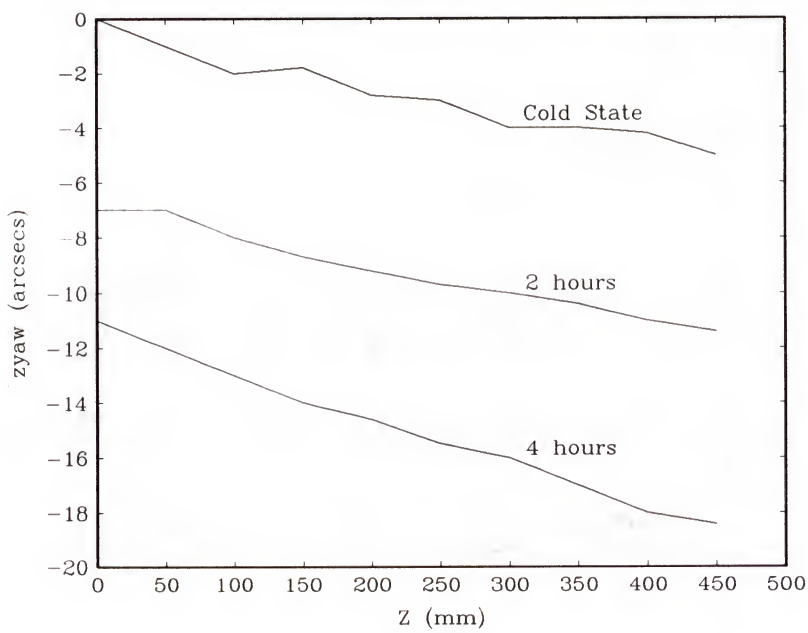


Figure 2-7 Thermally variant z yaw error of a two axis turning center.

CHAPTER 3
LITERATURE REVIEW

Error compensation techniques in machine tools have been studied since the late 1960s. Error compensation, according to Blaedel [4], can be defined as a technique to cancel the effects of repeatable errors by either directly or indirectly measuring the errors or by predicting them using a model previously established for the process. Most of the early work concentrated on acceptance checking and qualitative characterization of machine tools. Tlusty [5] described how a system of tolerances can express the accuracy of numerically controlled (NC) machines. A set of necessary measurements for acceptance checking of NC machines was also provided. These measurements were in the form of selected lines within the workspace of the machine.

Leete [6] developed a method for calculation of unwanted rotations, displacements and squareness errors of the machine slides. The translations and rotations of a slide were measured using two displacement measuring sensors mounted against a beam of light that acted as a perfect straightness reference. The outputs of the displacement sensors were converted into proportional phase shifts. These phase shifts were then used to modify the servo signals. Similar techniques were adopted by Wong and Koenigsberger [7] to compensate for slide misalignment

errors.

Goodhead, McGoldrick and Crabtree [8] used a precalibrated, nonperfect straightness reference, as opposed to a perfect straightness reference, in order to measure slide misalignment errors. They used fine wires fixed between two points on the machine frame located in the direction of slide movement under calibration. By using geometrical relationships similar to Leete, they calculated the resultant errors along the machine axes caused by slide misalignments. The concept of using reference straightedges was also applied by Bryan and Carter [9] to design a coordinate measuring machine (CMM). These straightedges were mounted on a separate metrology frame inside the machine bases, and LVDTs were used to measure the straightness errors. These straightness errors were then compensated during the CMM operation. Koliskor et al. [10] measured profile errors of machined parts and then generated modified NC tapes to compensate for measured errors in all future parts machined by the machine tool.

All of the above mentioned error compensation techniques concentrated on reducing individual components of geometric errors. They did not account for thermal growth of the machine tool and were fairly expensive and impractical to be adopted for many machining applications. Parallel to these efforts, there were attempts to model machine tool errors by many researchers, including Schultschick [11], Hocken et al. [12], Koval and Igonin [13], Portman [14] and Love and Scarr [15]. All of these efforts were successful in reducing errors in machine tools in the cold-state and clearly demonstrated the viability of preprocess calibration techniques.

The importance of thermal error compensation in machine tools received little attention from the machine tool industry before the 1980s. However, the importance of testing and evaluating thermal deformations of machine tools were identified by McClure [16] in 1969. Thermal distortions on machine tools were described as linear dynamic processes. Distortions in the machine tool structure caused by multiple internal heat sources are approximated by a linear lumped model. The parameters of the model are identified by experimental data, and thermal distortions are compensated.

Tlusty and Mutch [17] demonstrated typical features of thermal deformations in machine tools by computing transient thermal fields and transient thermal deformations of simple machine structures. They provide rules to identify the error sources that need to be measured and also the techniques to make these error measurements in order to characterize thermal deformations of machine tools for a wide range of thermal states. They also showed that thermal time constants for heating and cooling are different in machine tool structures.

Okushima, Kakino and Higashimoto [18] conducted a study to reduce spindle displacements caused by temperature variations around the spindle head. A model based on the correlation between the temperature variations and spindle displacements was developed. This model was then used to predict the spindle displacement. Compensation was achieved by injecting the corrections into the position feedback loop through an analog circuit. Sata, Takeuchi and Okubo [19] used external heat sources to shorten the time needed for warming up the machine

to thermal equilibrium. This helps in reducing the variations in the machine temperature distribution and hence stabilizes the deformations in the machine.

A new trend developed in the late 1970s towards building accurate and thermally stable machines. The diamond turning center developed by Donaldson [20] at the Lawrence Livermore Laboratory is a prime example of these efforts. In order to reduce thermally induced errors, the turning center was designed to include air bearings for the spindles and slides, low thermal expansion graphite material for machine structural elements and cooling systems for the spindle and the environment. These modifications reduce thermally induced errors to below 25 microns. Woytowitz et al. [21] and Wills-Moren et al. [22] used air bearings and magnetic bearings, respectively, for the spindle and guideways of high precision machines in order to reduce friction. Furukawa et al. [23] designed a ceramic guideway for reducing guideway friction. There has also been a recent development at Lawrence Livermore Laboratory to construct temperature controlled enclosures that surround the whole machine. They found these enclosures to be superior to temperature controlled rooms because they consumed less energy, saved space and were cheaper to maintain.

There have also been many efforts to use modern materials to design components of the machine tool. A new material called CARAT (coating aging resistant aluminum) was used in CMM products manufactured by Zeiss Company. This alloy helps in rapidly equalizing temperatures due to its high thermal conductivity, thus reducing temperature gradients. This was demonstrated by Placek

[24]. Other materials with low thermal expansion, such as graphite composites, concrete and alumina-ceramic, have also been studied for machine tool applications.

Takada and Tanabe [25] used passive heat control techniques in which insulation pads where used to control the heat flow between the machine components and hence control the thermal deformations in the machine. Nishiwaki, Takeyama and Hori [26] used an external heat source in the form of a heat pipe to create a symmetric temperature distribution around the machine structure and thus decreased the thermal deformation/bending of the machine structure. Spur et al. [27] directly heated the spindle and its housing in order to minimize the warm-up time required for stabilizing thermal deformations.

Weck and Zangs [28] and Venugopal and Barash [29] used the finite element method (FEM) to evaluate deformation characteristics for machine tools by considering a variety of designs for the machine tool temperature fields. Experimental data collected on the machine tool was then used to evaluate the accuracy of the FEM model. Due to the complicated nature of the heat sources in machine structures, it is very difficult to conduct a satisfactory heat transfer model to predict thermal errors. Hence, FEM techniques are thought to provide no more than predictions of qualitative trends. Also, implementing FEM methods in real-time is believed to be a major problem since the time required to carry out the necessary calculations for even simple structures is beyond the limits of real-time error compensation [33]. In a recent survey on the international status of thermal research, Bryan [30] reported that FEM techniques could predict only 20% of the

thermal distortions in machine tools.

Moriwaki [31] used finite difference methods to predict spindle growth of a turning center. A piezoelectric crystal based positioning mechanism adjusted the tool position based on the compensation provided by the finite difference method. Janeczko [32] developed a software approach to compensate for spindle growth. A CNC controller stores an empirical model for the spindle growth. Temperature measurements around the spindle were input into the model, and compensation signals were generated to offset the spindle growth. The novelty in this approach was that compensation was achieved entirely through software. This approach is faster, simpler and cheaper than the hardware method.

An important contribution to real-time error compensation in machine tools was made by Donmez [33] in 1985. In this approach, the machine is viewed as a kinematic chain in which each joint nominally possesses a single degree of freedom along its intended direction of motion. An "error motion," a small, general, rigid body motion caused by imperfect construction of the joints, results in actual motions of the body that differ from the intended motion. The error motion of each joint had six degrees of freedom (three translations and three rotations) defined with respect to the three coordinate axes of the joint frame. An expression for the total error at the tool point (with respect to the workpiece) was derived by using techniques of spatial rigid body kinematics (4 X 4 homogeneous transformation matrices) to describe the error motion of each machine component. These expressions contained the six component errors of each axis of motion, with each error component being

a scalar function of its joint position. In addition, the squareness errors among the three coordinate axes of the machine were also included.

Donmez [33] used a laser interferometer system to measure the scale and yaw errors of a two axis turning center at many points along the joint travel. Capacitance gages were used to measure straightness and squareness errors. An empirical model of the machine was constructed using regression models. This model also correlated temperatures around the machine tool in order to evaluate thermal deformations. A single board microprocessor with several interface boards was developed as the compensation controller. The controller calculates the total error at the cutting tool using the empirical model and feeds the correction value to registers of the CNC controller. This compensation occurs within 20 milliseconds and thus has the advantage over NC tape modification in performing a contour machining compensation without any hardware modification to servo-drive loops of the machine.

Zhang [34] and Teeuwsen et al. [35] developed similar models for CMMs to compensate for geometric and thermal errors caused by variations in the environmental temperature. Kurtoglu [36] and Anjanappa et al. [37] compensated for geometric errors of a milling machine using the kinematic model developed by Donmez. The methodology developed by Donmez provided a general mathematical framework to measure and empirically model all the parametric errors of a machine tool and their thermal variations. However, this method is time consuming and expensive and requires skilled technicians to use the calibration equipment. These limitations are the reasons for the reluctance in the industrial community to adopt

this technique.

Parallel to these efforts, a direct method of error measurement was being considered by many researchers. This method used artifacts or standards with known dimensions to measure the error at the cutting tool directly. The standards or artifacts are measured by the machine, and the difference between the measured results and the known lengths are used to calculate the total error at the cutting tool. Research in this area can be found in the work by Kunzmann et al. [39-40] and Neumann [41]. They used artifacts or standards of fixed shapes and known dimensions such as ball plates, gauge blocks, cubic boxes, tetrahedrons, etc. Jouy [42] used the invariant properties of metric tensors in a curvilinear coordinate framework in combination with an artifact to derive the scale and angular errors of a CMM. Ferriera and Liu [43] developed an artifact that contained a grid of cubes held together by wires and used it in conjunction with assumed functions for the individual error components to compensate for thermal variations in the geometric errors of a machining center. While these methods have the advantage of closely replicating real measurements, they have the disadvantage of difficulties in manufacturing and maintaining reference standards or artifacts.

There were other attempts to replace fixed-shape standards or artifacts by kinematic reference standards, including Bryan [44], Knapp [45] and Woollett [46]. They used a ball bar and other kinematic standards such as the machine checking gauge (MCG) patented by Renishaw to simplify the measurement of error at the cutting tool. These standards have a kinematic arm of known length that pivots on

one end and can change its orientation within some angular range. While data acquisition is simplified in comparison to fixed-shape standards, the key limitation to this approach is that it is not possible to cover the entire working zone of the machine tool with its fixed length. Another limitation in these approaches is the inability to provide any information about the individual error components of the machine tool. Further, these methods do not account for thermal growth in machine tools.

In addition to these methods, research was conducted in other areas that influence thermal errors. Two key studies included the correlation of machine tool inaccuracy directly to the environmental variations. McClure [47] improved on the accuracy of manufactured parts by upgrading the ambient-temperature control of the room that housed the machine tool. Hamar [48] showed that painting the windows of the room that housed the machine tool eliminated the positioning problems of the machine tool. The success of this approach was attributed to preventing sunshine into the room that housed the machine.

Chen [49] extended the work of Donmez to a three-axis machining center. A set of 32 error components were synthesized into the time-variant error model of the machining center instead of the well-known 21 component error model of a 3-D machine. This is due to the inclusion of nonrigid kinematic effects and tool length changes into this model of the machining center. A comparison was made between the predicting abilities of empirical machine models constructed by multiple regression analysis and neural network models. In addition to this, an on-line

method of data collection was developed to measure nonrigid body kinematic effects of the machining center. A compensation controller based on an IBM/PC was developed and linked with a DynaPath Delta20 CNC controller to compensate for the total error at the cutting tool in real-time. Dimensional errors of a cut part were reduced from 92.4μm to 18.9μm in a dimension of 404 X 310 mm^2, and the depth difference of milled surfaces was reduced from 196μm to 8μm. While this method is more comprehensive than that proposed by Donmez, it still suffers from the same drawbacks that have caused a reluctance to adopt it by the industrial community.

Soons, Theuws and Schellekens [50] developed a generalized error model for multiaxis machines that accounts for errors due to inaccuracies in geometry, finite stiffness and thermal deformations of the machine's components. An empirical model was constructed using calibration data collected by an artifact-based procedure. They identified significant parameters that were used to provide compensation values for a five-axis milling machine and a three-axis coordinate measuring machine. Hatamura et al. [51] employed thermal actuators to compensate for thermal deformations in a machining center actively. The machining center also incorporates force and deformation sensors to provide a fail-safe mechanism and to protect from a catastrophic failure. Errors were controlled to a level of 10μm, approximately one third of the level of the uncontrolled machine. Jedrzejewski and Modrzycki [52] modeled the thermal behavior of a machine tool on the determination of power losses in the kinematic system components, temperatures and thermal displacements. Matsumura et al. [53] developed an adaptive prediction

system of machining accuracy with a neural network. Machining features are extracted from several operations and identified using a fuzzy integral model of an expert engineer to decide input information of a network. This approach was shown to predict dimensional accuracy for some machining operations.

The techniques described above have been implemented in many CMMs and to a lesser extent in machine tools. The results are generally satisfactory, especially for CMMs operated in controlled environments where thermal effects are negligible and cutting loads do not exist. However, this approach is less applicable to machine tools. The expense of special instrumentation necessary to make the required measurements is substantial, equaling or exceeding the cost of all but the largest and most sophisticated machine tools. Even if instrumentation is available, specially trained technicians are needed to make the measurements. Measurement of each parametric error of a machine tool requires thermal duty cycles of 8-12 hours. Therefore, data collection times required to create thermal models are measured in weeks. This expense and difficulty have effectively prevented adoption of these techniques in industry. The neural network approach proposed in this research offers the possibility of improvement in accurate machining for a wide range of machine tools and CMMs without design changes or modifications in mounting.

Motivation

Based on the literature survey, it can be seen that most research mitigated thermal errors in one of three different ways. The first method minimizes the

problem by thermally stabilizing the machine. This is done by warming up the machine to thermal equilibrium before using it. This method is old and was used by the industry as early the 1950s. In a survey conducted by McClure [16] at a turbine plant in Westinghouse Corporation's Steam Division, the operator adjusts his work pattern to accommodate thermal influences on the machine by allowing time for "warmup" and then working late in the day. This method is not economical for industry as it requires a long period of downtime before the machine can be used. The second method uses design changes to mitigate effects of thermal errors in machines. This is accomplished either by using modern materials such as CARAT to design machine components or by isolating the major heat sources from the positioning elements. In some cases, these changes are complemented with a thermally controlled environment. While this method can be adopted for new machines, it is neither cost-effective nor practical to retrofit existing machines. Techniques that use special materials for machine components can also be very expensive to implement on new machines.

The third method attempts to build an empirical model that predicts thermal errors of the machine and to compensate for them. The model is developed by using the time-variant characteristics of individual error components of a machine measured at various thermal states of the machine. These measurements are typically done by running the machine continuously for several hours and periodically measuring the individual components, until the machine attains thermal equilibrium (i. e., temperature readings on the machine reach a steady state value). The machine

is then stopped and allowed to cool, periodically measuring individual components, until the machine attains cold-state conditions. Since machine tools are typically massive with high heat storage capacity, the time required for a thermal duty cycle may range from 12 to 16 hours (which includes both heating and cooling periods). The main drawback in this approach is that the empirical model can be built only after each component of error is characterized for all the thermal states. For example, a three-axis machine would require approximately 250 hours (i.e., 6[errors/axis] x 3[axes] x 14[hours]) to characterize the thermal errors of the machine completely. Furthermore, it is difficult to ensure identical starting cold-state thermal conditions (while characterizing the individual error components) in a typical shop-floor environment, as it is usually not temperature controlled. Hence, this method, while accurate, is time consuming, uneconomical and infeasible for most machine tool users. The principal motivation for this work is to overcome the above limitations in current techniques for machine tool calibration.

Research Objective

The main objective of this research is to provide a fast and accurate method to compensate for geometric and thermal errors of a turning center. This is to be achieved by a direct workspace identification technique [49] in which the total error at the cutting tool is measured directly using a new measuring device called the Laser Ball Bar (LBB) which was designed and tested at the University of Florida by Mize and Ziegert [54-55]. This device obviates the need for measuring the individual error

components of the machine and hence cuts down substantially on the data acquisition time. Furthermore, it offers the advantage of closely replicating real measurements. Since the distance measuring technique is based on laser interferometric principles, the LBB is highly accurate. The working principle of the LBB will be outlined in more detail in the next chapter.

While the setup and data acquisition time for the LBB based error measurement process is minimal, it is not desirable to interrupt the machining process on a regular basis for calibration purposes. In order to overcome this problem, a model of the machine needs to be built that can make correct predictions of the positional errors for a wide range of operating conditions characterized by noisy and unpredictable environments. A second goal of this research is to build such a model using the learning and decision-making abilities of neural networks.

There has been considerable interest in the use of neural networks for a variety of manufacturing applications, including learning of process models and adaptive control [56-59], monitoring and diagnostics [60-64] and quality control [65-68]. These researchers utilized the favorable features of neural networks, such as parallelism, robustness and compactness, in order to shorten the reaction time of systems, increase the product quality and make systems more reliable. Kalle and Ziegert [69] studied the feasibility of using the back propagation (BP) neural network developed by Rumelhart and McClelland [70] for machine tool calibration. A kinematic model of the machine was simulated by assuming realistic functions for the individual component errors of a two-axis turning center. The BP neural network

was then trained to learn the error map of the machine tool. Further, it was shown in simulation that a considerable improvement might be achieved when a part is machined with error compensation provided by the BP neural network compared to machining without compensation. This work was then experimentally tested by Srinivasa, Ziegert and Smith [72] by measuring the individual error components of a MAZAK QT 28N turning center using the laser interferometer and capacitance gages. These errors were used to build a kinematic model of the machine tool, which in turn was used to generate the training data for the BP neural network. After training, the neural network was able to predict accurately the error vectors at any point in the workspace of the machine. Further, the BP neural network was shown to learn and predict geometric errors and their thermal variations in Srinivasa, Ziegert and Smith [72].

Chen [49] at the University of Michigan also used the BP network to learn spindle thermal drifts as well as geometric errors and their thermal variations with good success. These results show that the BP neural network is capable of approximating time-variant machine tool error maps. However, this network requires prohibitively long training times. The network architecture (i.e., number of nodes in each layer) and learning parameters must be decided on a trial and error basis. While the algorithm can be clearly stated, the details of the computing carried on by the network are not easily accessible, and the effects are not easily traced. Furthermore, this network is only capable of on-line use after all of the training data are collected. These drawbacks coupled with the lengthy data acquisition procedure

make it difficult to implement calibration in a timely and efficient manner.

The proposed research will use a new approach for learning and prediction of thermal errors of a machine tool. A modified version of the Fuzzy ARTMAP neural network originally developed by Carpenter et al. [73] is to be used. This network is capable of incremental learning of analog multidimensional mapping. Using the network, the problem is formulated as one of pattern recognition. An input pattern of nominal spatial coordinates (measured by axis position encoders of the machine tool) along with temperature readings at strategic locations on the machine(measured using thermocouples as sensors), and positional errors (measured by the LBB) is presented to the network. The problem of approximating machine tool error maps is converted into one of mapping the nominal coordinates of the machine axes and temperature readings of certain internal heat sources into positional errors represented by discrete classes. These classes are determined based on the minimum resolution allowable by the machine controller for each axis. This is justifiable as increments smaller than this cannot be executed by the controller. For example, if the minimum resolution of an axis is 1 micrometer and the largest positioning error component for that axis as measured on the machine is 100 micrometers, then the neural network will associate any input pattern, nominal coordinates of the machine and temperature readings into one of 100 discrete positional error classes.

Viewed in this sense, the problem is similar to character recognition where each character is classified into a class that is separated from other classes by some

criterion such as Hamming distance [74] or Euclidean distance [75]. The Fuzzy ARTMAP does not suffer from the drawbacks of the BP neural network. The associative mapping is refined as more training examples are presented to the network. Multiple training passes are not needed, so long learning times are unnecessary, and real-time learning is feasible. The working principle of the network is very easy to understand. Another important feature of the Fuzzy ARTMAP neural network is that it can accommodate new training data without having to relearn the entire training set. In addition, the learning can be performed in an unsupervised fashion, thereby making it suitable for an automated factory environment.

Scope of Research

In this research, a direct method of machine calibration is used to learn and predict the geometric and thermal errors of a machine tool using a neural network and the LBB combination. The neural network to be used is called the Fuzzy ARTMAP [73] and is capable of incremental supervised learning of multidimensional maps in response to arbitrary sequences of analog or binary inputs. Preliminary work [76] has shown that the Fuzzy ARTMAP neural network can learn machine tool error maps in real-time and with adequate accuracy in a single pass of training data.

The training set of the neural network will consist of the set of cutting tool locations along with a set of current and past thermocouple readings measured at strategically located points on the machine as inputs, and the corresponding tool point error vectors as measured by the LBB as outputs. These inputs and outputs

are similar to those used by Donmez [33] and Chen [49], who have numerically correlated the predicted total error with the readings of temperature probes and generated appropriate positional compensation values. The data collected during the cold-state of the machine corresponds to the geometric errors of the machine. The errors collected at all the other thermal states are due to the thermal distortion of the machine tool error map.

An interesting aspect of learning using the Fuzzy ARTMAP is that it occurs as and when the data are collected. Also, the network architecture is very easy to use, has a small number of parameters, and requires no problem-specific crafting or choice of initial weight values. If proven feasible, this approach would represent a method for dealing with thermal effects on machine output in real-time. Using the LBB, all necessary input data for training can be collected in a single thermal duty cycle. The data collection cost as well as time taken to collect and learn would be minimal, thus making it economically viable for the machine tool industry.

The effect of elastic errors has been well studied by Tlusty and Smith [77-79] at the University of Florida and thus will not be included in this research. A case study by McClure [16] shows that workpiece and cutting tool expansion during machining can be reduced by more than 75% when a coolant is used during the machining process. Hence, the thermal effects due to the cutting process will also not be studied in this research.

An important component of thermally induced errors in machine tools is error associated with the spindle. A novel approach has been developed to measure the

thermal drifts and tilts of the spindle using the LBB. The data required to extract the spindle errors will be collected concurrently with the measurement of thermally induced changes in geometric errors (i.e., the thermal error map). These errors will then be transformed into the total error at the cutting tool and included in the inputs to the Fuzzy ARTMAP.

The Fuzzy ARTMAP network will be simulated on a digital computer by developing appropriate software. Experiments will be performed on the MAZAK QT 28N turning center housed at the Machine Tool Research Center. The trained neural network will be tested experimentally by comparing the results of actual measurement of tool point errors at various locations not included in the training set with those predicted by the Fuzzy ARTMAP for various thermal states under realistic machining conditions. A real-time error compensation scheme for the MAZAK turning center was developed by Sumanth [80] based on the research conducted by Yee and Gavin [81] at the National Institute of Standards and Technology at Gaithersburg, MD. The outputs of the Fuzzy ARTMAP neural network will be interfaced with this error compensation system. Actual machining of parts will be conducted with and without compensation in order to evaluate the performance and potential advantage of this approach. The scope of this research can be summarized as follows:

1) Demonstrate the ability of the Fuzzy ARTMAP neural network to learn and predict geometric and thermal errors from training data of the tool point error vectors, cutting tool location and strategically located temperature probes. The tool

point error vectors will be measured using the LBB.

2) Demonstrate the ability of the Fuzzy ARTMAP neural network to learn and predict spindle drifts from training data consisting of spindle drifts, temperatures from strategically located temperature probes and spindle speed. The spindle drift errors will also be measured using the LBB.

3) Interface the outputs of the Fuzzy ARTMAP neural network with an error compensation system developed by Sumanth [80].

4) Evaluate the performance of this approach by actual machining of parts with and without error compensation.

CHAPTER 4
ARTIFICIAL NEURAL NETWORKS

An Overview of Neural Networks

A neural network according to Nielsen [82] is defined as a parallel distributed information processing structure. It is composed of processing elements that are interconnected by signal channels called weight connections. These processing elements (also called artificial neurons) have a transfer function and may have a local memory analogous to a rudimentary sense to the biological neuron. The weight connections have varied strengths based on the learning experience of the network.

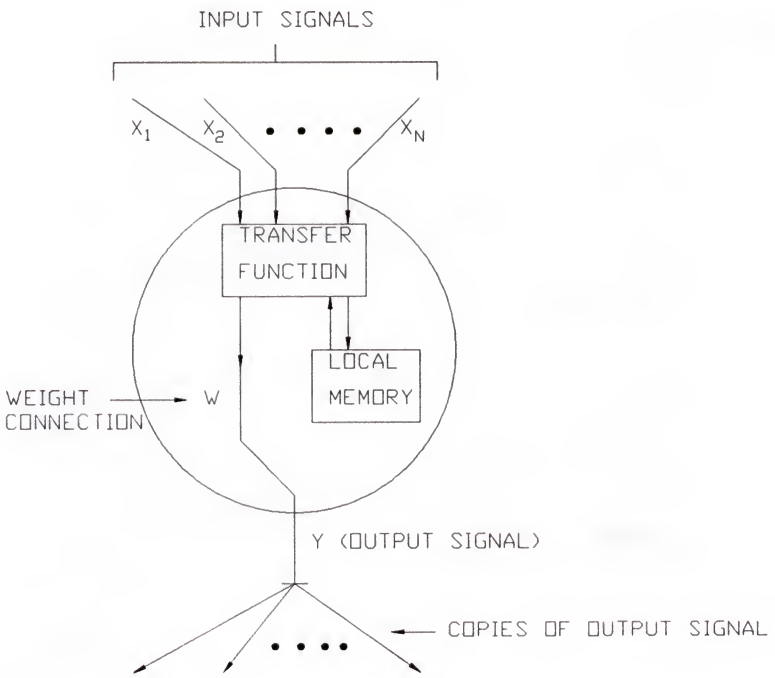


Figure 4-1 A typical artificial neuron/processing element.

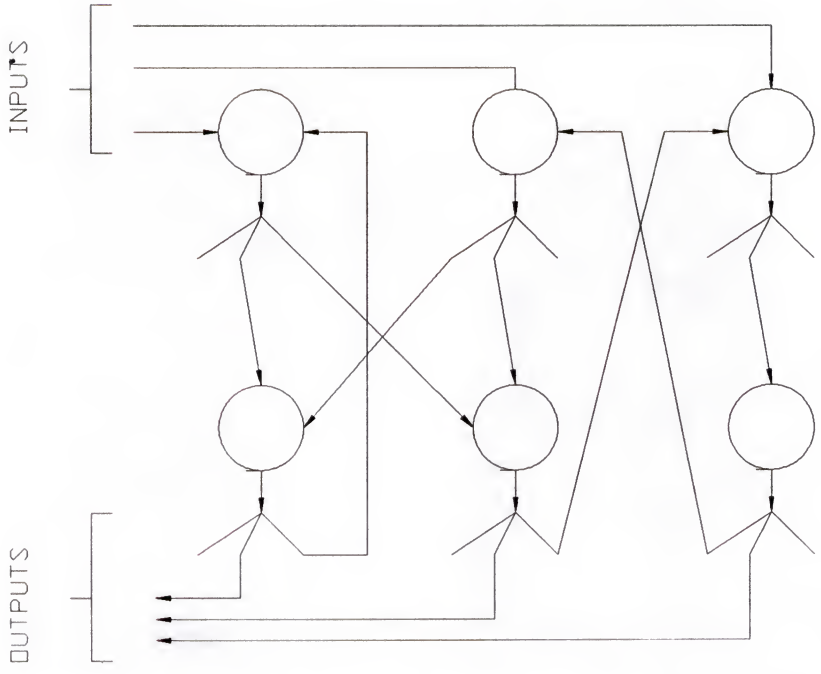


Figure 4-2 A typical neural network architecture.

A typical processing element or artificial neuron is shown in Figure 4-1. It receives input signals arriving via incoming connections. Given these inputs, the transfer function helps in providing the processing element's output signal y. This output signal y is multiplied by a weight value and branched into copies after leaving the processing element. A typical neural network architecture is shown in Figure 4-2. This figure shows inputs (viewed as a data array x) entering the network from the external environment and outputs (viewed as a data array y) leaving the network to the external environment. The general functional form of a neural network is analogous to a software subroutine or procedure (input → processing → output). This observation forms the basis for embedding neural networks into programmed computing systems.

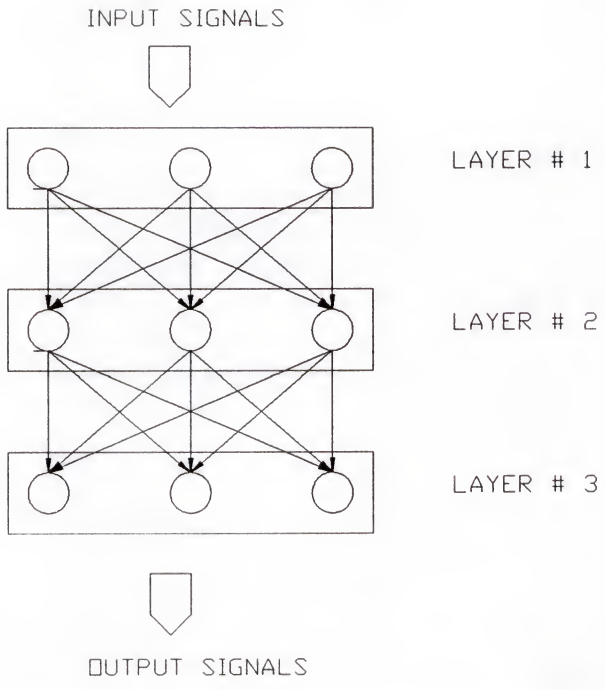


Figure 4-3 A layered architecture of a neural network.

Neural networks have their processing elements divided into disjoint subsets or groups called layers as shown in Figure 4-3. The processing elements within each layer essentially have the same transfer function and can send connections to other processing elements on the same layer, as well as to ones on other layers. The outputs of all the processing elements within a layer represents the activity pattern of that layer.

Most neural networks include an additional layer, called an input layer in which each processing element receives exactly one input, which arrives from the outside world. The processing elements of the input layer typically have no function other than to distribute the signals impinging upon them to other processing elements of the network. They have no local memory and the transfer function is to simply

pass on the signals without any changes. Processing element transfer functions usually have a subfunction called a learning law. This law is responsible for adapting the input-output behavior of the processing element transfer function in response to the input signals that impinge on the processing element. This is achieved by changing the values of weights that connect the processing element's output to other processing elements.

Different neural network architectures can be realized by changing the type of transfer function of each processing element, the learning law and/or the number of layers. However, neural networks are commonly classified into three classes based on the learning method adopted. The three classes are: supervised learning, reinforcement learning and unsupervised learning. In supervised learning, a teacher from the external world provides the desired response at each time step during the learning process. In reinforcement learning, the teacher's response is not direct, immediate, and informative as in supervised learning and serves more to evaluate the state of the system. The presence of a teacher or a supervisor to provide the correct control response is not assumed in unsupervised learning. The BP neural net [70], the functional link net [75], Boltzmann machine [83] and multilayered perceptrons [74] are examples of neural networks that use supervised learning. Reinforcement learning algorithms can be found in the work of Barto [84], Mendel [85], Gullapalli [86] and Berenji [87]. The adaptive resonance theory (ART) network [88], Kohonen's network [89], Anderson's network [90], and the adaptive bidirectional associative memory (ABAM) network [91] are examples of neural networks that

make use of the unsupervised learning paradigm.

Unsupervised learning networks are attractive since they do not need to be trained by an external agent before learning can proceed. The Kohonen and Anderson networks work well with both binary and analog inputs but cannot learn new associations once they are trained. The ABAM network, while conceptually simpler than the ART network, is subject to pattern orthogonality constraints and pattern storage problems [92]. In contrast, the ART network does not have the above limits on its performance. The processing of inputs in real-time is possible through fast learning rules used by the network. The ART model is also capable of directly recognizing a pattern without having to search provided that pattern has already been learned previously. This is comparable to the human ability to rapidly recognize familiar objects.

While the ART architecture has these interesting properties, it is only suitable for unsupervised clustering of binary inputs (features with 0 or 1) into recognition codes and not multidimensional function approximation. Computations from Fuzzy set theory [94] were incorporated into the ART architecture by Carpenter et. al [93] (called Fuzzy ART) in order to facilitate learning both binary and analog inputs (features with values between 0 and 1). Recently, a pair of fuzzy ART networks were coupled into a new network called the Fuzzy ARTMAP neural network [73] that is capable of real-time approximation of functional mapping. The term "Fuzzy ARTMAP" refers to the map between the two Fuzzy ART networks within it. This network has a simple architecture, learns on-line, and does not require problem

specific crafting of its parameters. The Fuzzy ARTMAP neural network can accommodate new training data without having to relearn the entire training set. In this research, the Fuzzy ARTMAP will be combined with the LBB to learn and predict thermal errors in machine tools.

Working Principle of the Fuzzy ARTMAP Neural Network

The Fuzzy ARTMAP neural network is capable of incremental learning of multidimensional maps in response to arbitrary sequences of analog input vectors. This network consists of two fuzzy ART neural networks called ART_A and ART_B as shown in Figure 4-4. During learning, ART_A receives a vector x^A, which is the input to the mapping, and ART_B receives a vector x^B, which is the correct output of the mapping to input x^A. These vectors are coded into recognition classes in their respective fuzzy ART modules and these classes are mapped to each other in real-time. In this section, the working principle of the fuzzy ART module will be described. For convenience, the process for the ART_A module only is outlined, as it is identical for ART_B module.

The ART_A module contains two layers of processing elements: layer F_1^A whose processing elements receive input signals from the external environment and a layer F_2^A that codes the outputs received from the processing elements of F_1^A into discrete classes among its processing elements. The processing elements of F_1^A do not have any transfer function. The input x^A contains M features denoted by x^A =

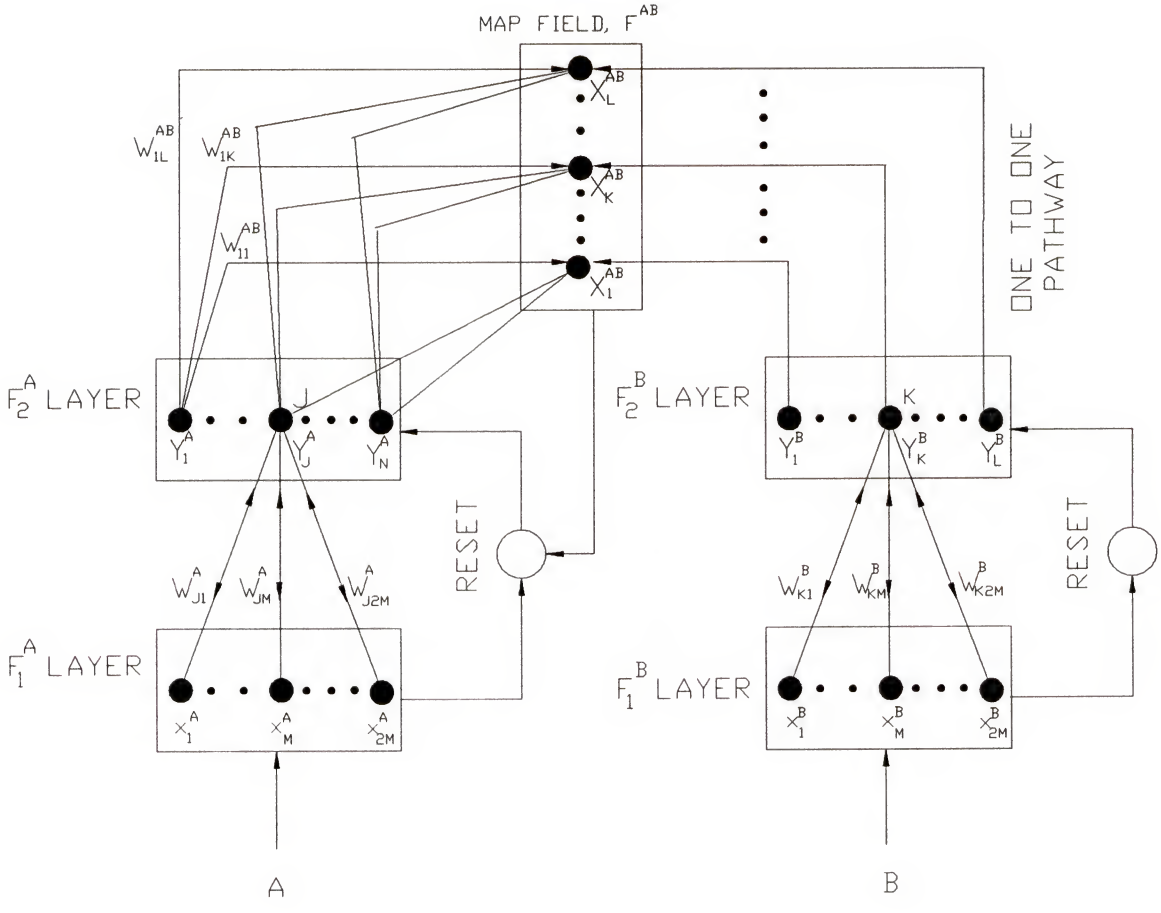


Figure 4-4 The Fuzzy ARTMAP architecture.

$(x_1^A, \ldots, x_M^A)$ where each x_i^A is normalized within the range [0, 1] range. The F_1^A layer further normalizes the input x^A across its 2M processing elements and produces an activity pattern $(x_1^A, \ldots, x_{2M}^A)$ across it. The processing elements with activity $(x_{M+1}^A, \ldots, x_{2M}^A)$ represents the complement of the first M features provided by input x^A where $x_{M+i}^A = 1 - x_i^A$ where i = 1, . . ., M. This normalization procedure is called complement coding [73, 93] and uses the inputs and its complements to unambiguously represent individual feature amplitudes. The outputs from the processing elements of the F_1^A layer are multiplied by the adaptive weight connections $w_j^A = (w_{j1}^A, \ldots, w_{j2M}^A)$ that connects a class j in F_2^A to the 2M processing elements of the F_1^A layer. Initially, $w_{j1}^A = \ldots = w_{j2M}^A = 1.0$ for j = 1 to N where N is the total number of processing elements in F_2^A layer. The weighted signals from the 2M processing elements of the F_1^A layer create an activity pattern $T_j(x^A)$ across each processing element j in the F_2^A layer where

$$T_j(x^A) = \frac{|x^A \wedge w_j^A|}{|w_j^A|} = \frac{\sum_{i=0}^{2M} \min(x_i^A, w_{ji}^A)}{\sum_{i=0}^{2M} w_{ji}^A} \tag{4-1}$$

where $\wedge$ represents the fuzzy AND operator [94]. The processing elements of the F_2^A layer compete with each other to determine the element J with maximum T_J activity as follows:

$$T_J = \max(T_j(x^A) : j = 1, \ldots, N) \tag{4-2}$$

The processing element J suppresses the activity of all other processing elements in

the F_2^A layer and creates an activity pattern in the F_2^A layer denoted by $y^A = (y_1^A, \ldots, y_N^A)$ where $y_J^A = 1.0$ and $y_j^A = 0.0$ for all $j \neq J$. The activation pattern across the F_2^A layer may be interpreted as "making a hypothesis" about the class that should represent the input x^A. The activation y^A prompts the F_2^A layer to check for an acceptable match with x^A by a condition called the vigilance criterion as follows:

$$\frac{|x^A \wedge w_J^A|}{|x^A|} = \frac{\sum_{i=0}^{2M} \min(x_i^A, w_{Ji}^A)}{\sum_{i=0}^{2M} x_i^A} \geq \rho^A \qquad (4\text{-}3)$$

The numerator of this equation $|x^A \wedge w_J^A|$ represents the learned expectation of F_2^A layer given input x^A and the ratio of $|x^A \wedge w_J^A|$ over $|x^A|$ represents the degree of similarity of w_J^A that represents the class of element J with input x^A. The parameter ρ^A is a dimensionless parameter called the vigilance parameter and can have a value between 0 and 1. Low value of vigilance leads to broad generalization and codes a wide range of inputs at F_1^A layer into each category J in the F_2^A layer. High vigilance values lead to narrow generalization and results in fewer inputs from F_1^A layer into each category J in F_2^A layer.

If the condition in equation (4-3) is not satisfied, then a reset occurs. This triggers a parallel search in the F_2^A layer among all the processing elements other than J for the next maximally active node J′. If the selected processing element J′ is already an established category for previous inputs x^A, and satisfies equation (4-3), then the weights $w_{J'}^A$ is appropriately refined to include the current input within the class represented by element J′. If the search ends by selecting a previously

untrained processing element J′, then the input x^A is learned and accommodated into the new class represented by element J′. Thus J′ is said to form the prototype for x^A in the F_2^A layer.

If the condition (4-3) is satisfied without reset, then a state of resonance/match is said to have occurred and the processing element J is said to have formed the prototype for the input x^A. The formation of the prototype J (or J′ as the case may be) is followed by a learning rule where the weights w_J^A are modified as follows:

$$w_{Ji}^{A(new)} = x^A \wedge w_J^{A(old)} = \min(x_i^A, w_{Ji}^{A(old)}) \quad for\ i = 1, \ldots, 2M \quad (4\text{-}4)$$

Equation (4-4) represents the fast learning rule [73] that makes on-line learning possible. Fast learning quickly adapts the inputs that may occur only rarely and that may require immediate learning from the network. The ability of humans to remember many details of an exciting movie is a typical example of fast learning. This learning is also stable because all the weights w_J can only decrease in time. The learning proceeds with familiar inputs directly activating their respective classes, while novel inputs trigger adaptive searches for classes that represent them better, until the memory capacity is fully utilized. Thus, inputs x^A and x^B presented to the F_1^A and F_1^B layers of ART_A and ART_B are coded within the F_2^A and F_2^B layers respectively.

The ART_A and ART_B networks are linked together via by a third layer F^{AB} called a map field. The map field forms predictive associations between the F_2^A and

F_2^B layers as follows. Let the activity pattern across the processing elements of the F^{AB} layer be denoted by $x^{AB} = (x_1^{AB}, \ldots, x_L^{AB})$ and let $w_j^{AB} = (w_{j1}^{AB}, \ldots, w_{jL}^{AB})$ denote the weight connections from the jth processing elements in F_2^B layer. Initially, $w_{j1}^{AB} = \ldots = w_{jL}^{AB} = 1.0$ for $j = 1, \ldots, N$. The processing elements of F_2^B and F^{AB} layers are connected by a 1-to-1 pathways (i.e., no adaptive weight connections): Let the inputs x^A and x^B be coded in element J of F_2^A and element K of F_2^B. The activity pattern $x^{AB} = y^B \wedge w_j^{AB}$ where $x_i^{AB} = \min(y_i^B, w_{Ji}^{AB})$ for $i = 1$ to L. If one of the processing elements x_i^{AB} is non-zero, then the class represented by processing element K of ART_B is said to predict the class represented by the processing element J of ART_A. If $x_i^{AB} = 0$ for all the processing elements i, then a mismatch event occurs and triggers a search in ART_A for a new processing element by increasing the vigilance ρ^A to a new ρ_{new}^A of ART_A as

$$\rho_{new}^A > \frac{|x^A \wedge w_J^A|}{|x^A|} = \frac{\sum_{i=0}^{2M} \min(x_i^A, w_{Ji}^A)}{\sum_{i=0}^{2M} x_i^A} \tag{4-5}$$

When this occurs, the ART_A search leads to activation of either an ART_A processing element that correctly predicts x^B (i.e., satisfies $x^{AB} \neq 0$ vector) or to a previously uncommitted processing element of F_2^A. Once a processing element J of ART_A learns to predict processing element K of ART_B, then $w_{JK}^{AB} = 1$ and $w_{jk}^{AB} = 0$ for $j \neq J$ and $k \neq K$. The vigilance parameter ρ_{new}^A is then reset to its original value ρ^A before the next pair of inputs (x^A, x^B) are presented to ART_A and ART_B networks.

In order to implement real-time learning using the Fuzzy ARTMAP neural

network on an IBM PC compatible, it was necessary to incorporate two changes in the algorithm. First, the number of nodes in the F_2 layer of ART_A and ART_B modules were recruited using dynamic memory allocation based on the need to classify an input-output pair into a new node(s) as opposed to pre-fixing this number before learning. This enabled the time for parallel search (in case of a reset) to be restricted to among the nodes that have already coded a pattern. If none of the currently recruited nodes can code the input-output pair presented, a new node (in ART_A, ART_B or both as the case may be) is created and it automatically codes the given pattern pair. Secondly, the weights of ART_A, ART_B and Map Field are stored in non-complement coded form as the complement set can be derived from the non-complement form. This enabled the reduction in memory required to store the knowledge in the form of weights. The Fuzzy ARTMAP neural network was programmed in Microsoft C on an IBM PC. The C code is listed in Appendix A.

CHAPTER 5
THE LASER BALL BAR

The Laser Ball Bar (LBB) is a linear displacement measuring device that was designed and tested by Mize and Ziegert [54, 55] at the University of Florida. It was developed to rapidly measure volumetric errors using trilateration methods. The design described in [54, 55] was primarily developed as a prototype to test the feasibility of using the LBB for machine tool calibration. A second generation LBB was recently designed by Mize in order to improve on the first design in terms of its working range, higher accuracy and ease of handling. The new LBB was calibrated at the Oak Ridge National Laboratory using the M60 CMM (believed to be the most

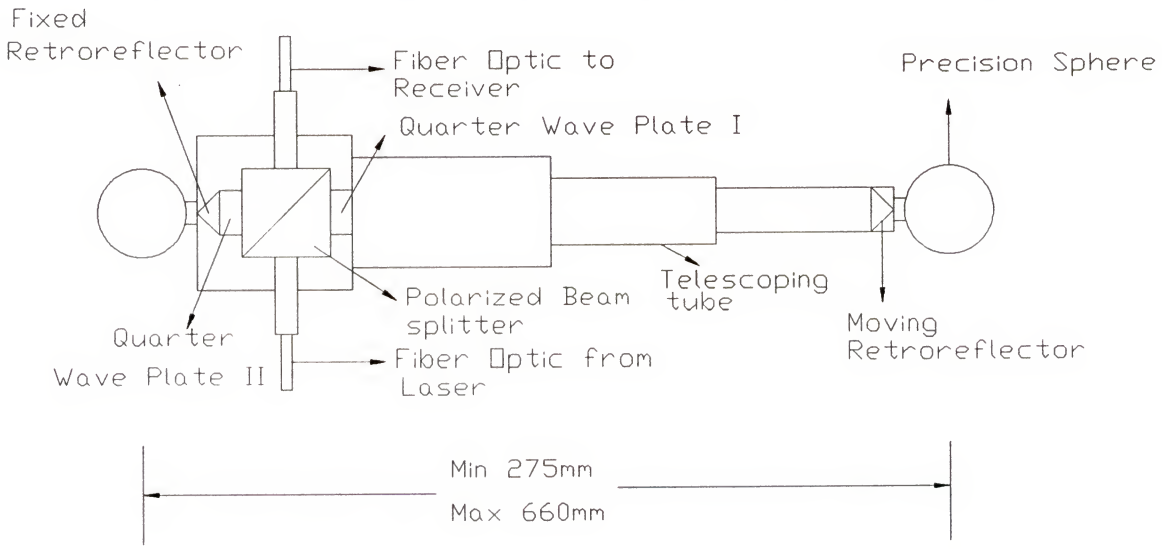


Figure 5-1 Schematic of the Laser Ball Bar.

accurate of its class in the world) and was found to have an overall accuracy of ± 1 micron [94]. In this research, the new LBB will be used to measure machine tool errors. This LBB consists of a laser interferometer aligned between two precision spheres by a telescoping tube as shown in Figure 5-1.

Wavelength stabilized laser light in the form of two orthogonally polarized beams (superimposed on each other) is received from the laser source to a polarization preserving fiber optic cable. This light is carried to the polarization beam splitter (PBS) where it is separated into two components based on its polarization. One component passes through the PBS to the fiber optic pick-up as the reference beam. The other component passes through quarter wave plate I and travels along the telescoping tubes to the moving retroreflector attached to the

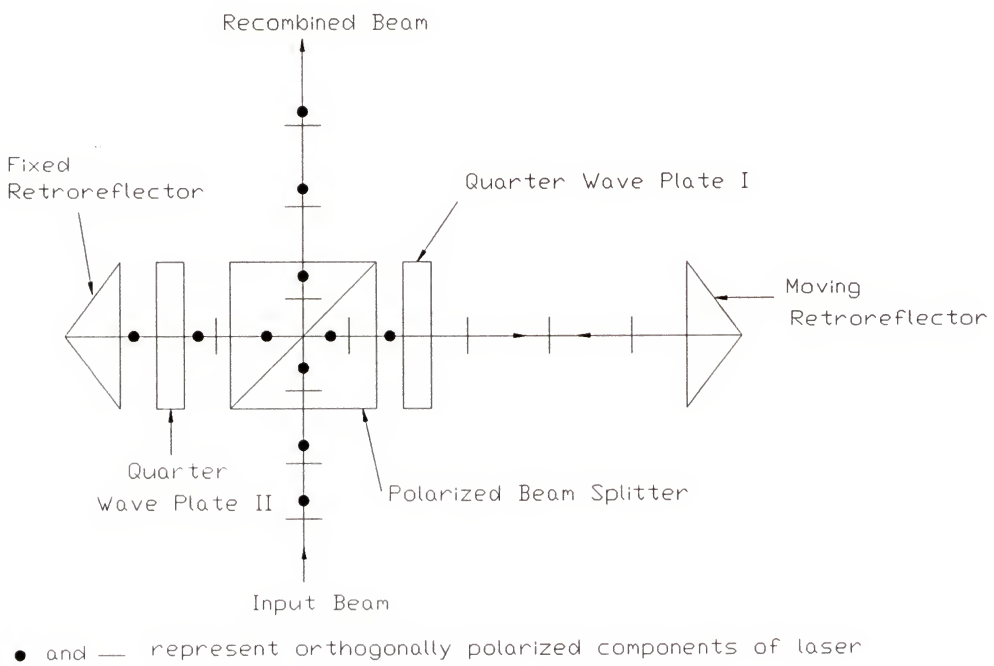


Figure 5-2 Path of the two orthogonally polarized beams during measurement.

opposite end. The retroreflector returns the beam collinear to the path of entry back back to quarter wave plate I. Since this passes through the quarter wave plate I twice, it becomes orthogonally polarized to its original state. As a result, this component passes through the PBS onto quarter wave plate II, the fixed retroreflector and again back through the quarter wave plate II. This component becomes orthogonally polarized to the state in which in entered the quarter wave plate II and hence is reflected by the PBS to the fiber optic pick-up and combined with reference beam. The path of light from laser to fiber optic pick-up at the receiver end is depicted in Figure 5-2. This recombined beam is carried to a receiver where the displacement information can be extracted from the phase relationship of their combined beat frequency relative to the reference frequency of the laser. For details on extraction of displacement information and the electronics associated with it, the reader is referred to [95]. Since the LBB is based on laser interferometric principles, it can only measure changes in length. However, in order to use the LBB for machine tool calibration, it is necessary to know its absolute length. This is accomplished through a self-initialization procedure described in a step-wise fashion in Figure 5-3. The LBB is placed between sockets #1 and #2 and the interferometer is zeroed. The end of the LBB in socket #2 is moved to socket #3 and the change in length recorded. The LBB is then placed between sockets #2 and #3 and is initialized to this length. Following initialization, the absolute LBB length is the initialization length plus any change in length recorded by the interferometer. The LBB used in this study has a minimum length of 275mm and an extension of 385mm.

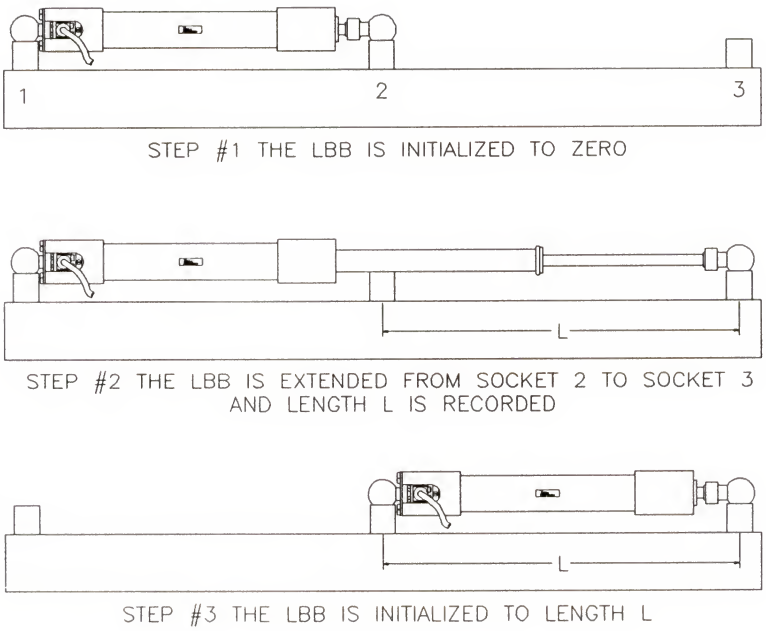

1
2
3
STEP #1 THE LBB IS INITIALIZED TO ZERO
L
STEP #2 THE LBB IS EXTENDED FROM SOCKET 2 TO SOCKET 3
AND LENGTH L IS RECORDED
L
STEP #3 THE LBB IS INITIALIZED TO LENGTH L

Figure 5-3 Self-initialization of LBB using the three socket technique.

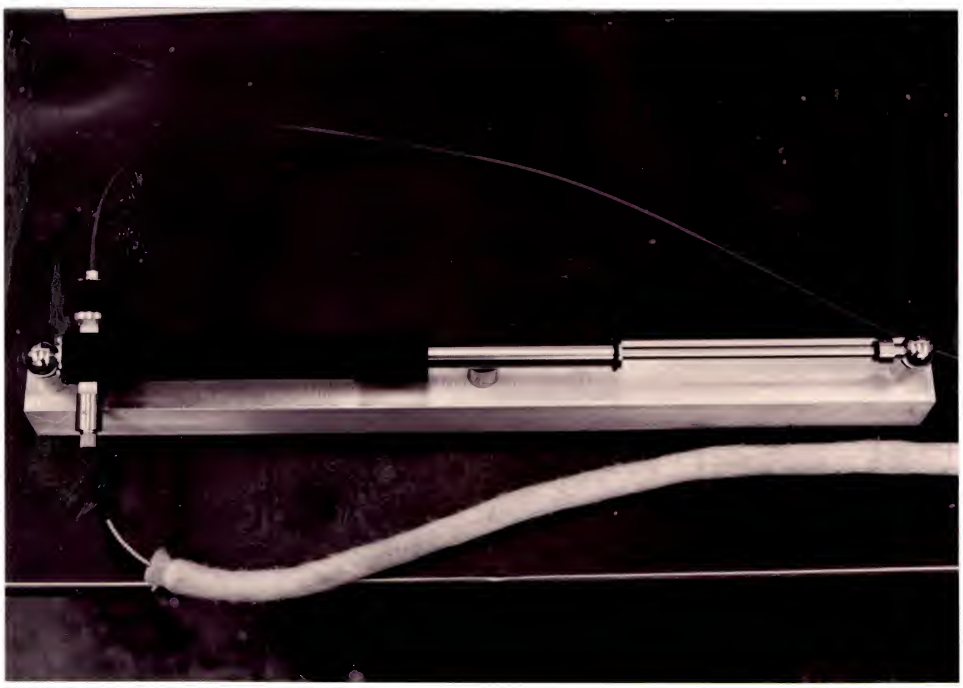


Figure 5-4 The LBB and the three socket initialization stand.

The LBB and the three socket initialization stand are shown in Figure 5-4. It should be noted that the time taken for initialization is extremely small and hence will not cause the distance between the three sockets to thermally grow during the process of initialization.

LBB Based Total Positional Error Measurement Technique

The positioning error of the cutting tool of a MAZAK QT 28N two axis turning center was measured using the LBB. The turning center consists of a bed which supports the headstock, tailstock and the carriage. The carriage supports the cross-slide and the cross-slide supports the turret. The headstock contains the spindle unit with a hydraulic chuck to clamp the workpiece. The tailstock is used for supporting flexible workpieces during machining. In this study, the tailstock is not used. The carriage motion is defined as the Z axis and the cross slide motion is defined as the X axis. A schematic of the turning center is shown in Figure 5-5.

The method of sequential trilateration is adopted to determine the position of the cutting tool relative to a reference coordinate frame. In order to use this method, it was illustrated by Mize [54] that three magnetic sockets in the plane of the turning center are sufficient to determine the total positional error map. However, a drawback in that approach is that it is not possible to measure the workspace close to the spindle where the workpiece is usually machined. Hence it was decided to use a four socket approach. In this approach, three magnetic sockets called the base sockets, are attached to the bed of the turning center and a fourth

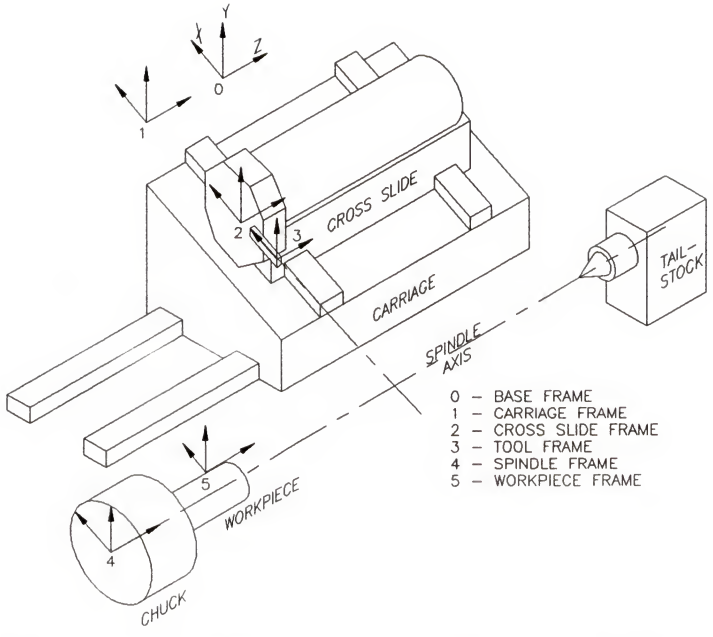


Figure 5-5 Schematic of the two axis turning center [54].

socket called the tool socket, is attached to the turret called the tool socket (i.e., where the cutting tool is mounted). These four sockets form a tetrahedron as shown in Figure 5-6a. The three base sockets define a base plane. The X_L axis is located along one leg of this triangle and the Y_L and Z_L axes are located according to a right handed coordinate system. The LBB is first initialized and is followed by measuring the base lengths of the tetrahedron (L_{B1}, L_{B2}, L_{B3}). Next, the LBB is placed between the first base socket and the tool socket. A CNC program moves the tool through a grid of points as shown in Figure 5-6b. At each grid point i, length L_{1i} is recorded. This procedure is repeated for the remaining two sockets to obtain L_{2i} and L_{3i} for all the grid points. The coordinates (x_i, y_i, z_i) of each grid point can be derived using (L_{1i}, L_{2i}, L_{3i}) and (L_{B1}, L_{B2}, L_{B3}) as follows (refer to Figure 5-6a):

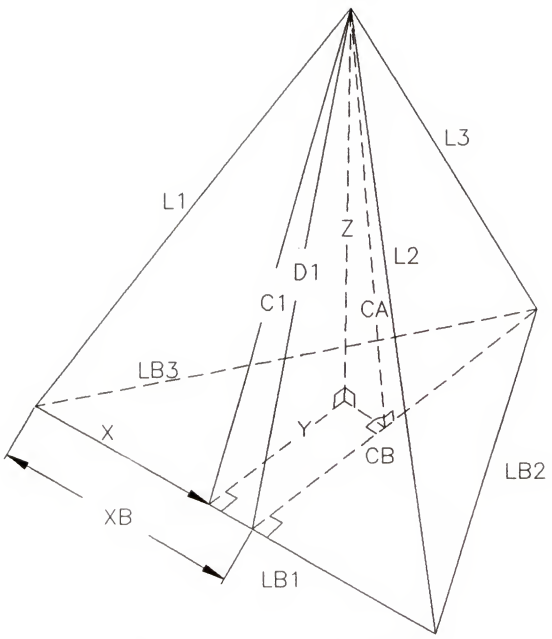


Figure 5-6a Tetrahedron formed by the LBB using the four socket approach.

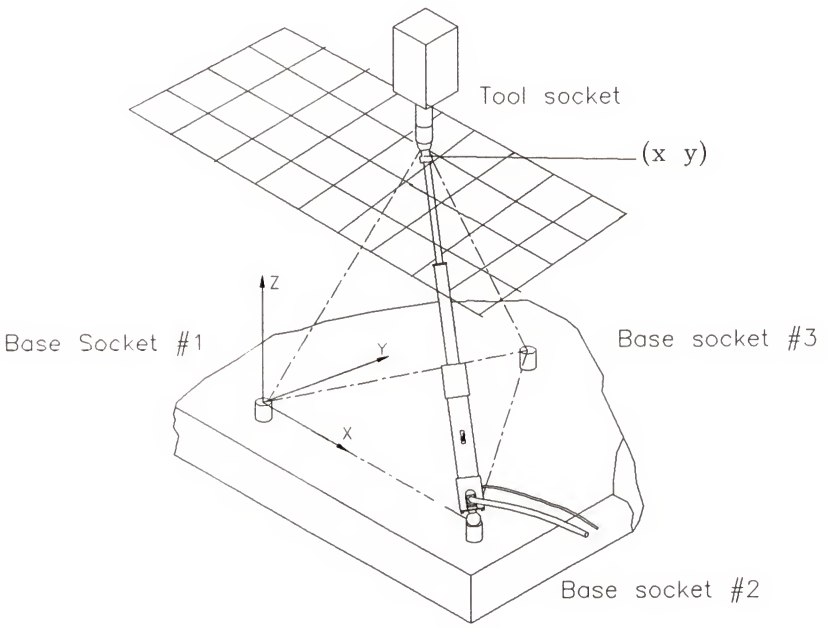


Figure 5-6b Sequential trilateration using the LBB.

$$x_i = \frac{L_{1i}^2 - L_{2i}^2 + L_{B1}^2}{2L_{B1}} \tag{5-1}$$

$$c_1 = \sqrt{L_{1i}^2 - x_i^2} \tag{5-2}$$

$$x_B = \frac{L_{B3}^2 - L_{B2}^2 + L_{B1}^2}{2L_{B1}} \tag{5-3}$$

$$c_B = \sqrt{L_{B3}^2 - x_B^2} \tag{5-4}$$

$$d_1 = \sqrt{c_1^2 + (x_B - x_i)^2} \tag{5-5}$$

$$y_i = \frac{d_1^2 - L_{3i}^2 + c_b^2}{2c_B} \tag{5-6}$$

$$z_i = \sqrt{c_1^2 - y_i^2} \tag{5-7}$$

The coordinates (x_i, y_i, z_i) are derived in the LBB coordinate frame (X_L, Y_L, Z_L). However, machining of workpieces is referred to the coordinate frame of the machine. If the base sockets are carefully aligned to be parallel with one axis of the machine and the plane forming the base is coplanar with the cutting plane of the machine, then the LBB frame will be parallel to the machine frame. When this is the case, the measured (x_i, y_i, z_i) coordinates represent the actual position of the cutting tool in the machine frame and hence can be directly used for error compensation. However, this technique is cumbersome and inflexible because it

requires precise alignment of base sockets to be parallel to an axis of the machine. This problem is overcome by deriving the transformation between the LBB coordinate frame and the machine coordinate frame as shown in Appendix B. Using the transformation matrix ${}_{B}T^{M}$ (as derived in Appendix B), the ball bar coordinates (x_i, y_i, z_i) are converted into the machine coordinates and compared with the nominal machine coordinates to obtain the total positional errors for all the grid points. In the case of a three axis machining center, it is easy to locate the three base sockets on the work table of the machine. However, it was difficult to find such a surface on the two axis turning center. Hence an I-beam was fabricated and attached to the bed of the machine as shown in Figure 5-7. Two base sockets were mounted on the ends of the beam. The third base socket was mounted on top of a socket fixture that

Figure 5-7 The I-beam and magnetic socket fixture attachments.

was fabricated and attached below the guideway of the tailstock. In order to test the stability of these attachments to machine vibrations and deflections, a case study was conducted. In this study, the LBB was mounted between each base socket and a socket attached to the clamped spindle. For each socket, the carriage was exercised to move back and forth. Since the spindle and base sockets are stationary, the LBB length should not change for the carriage motion. However, the LBB length increased by 2 and 7 microns when mounted between the spindle and socket #2 and #3 respectively. The LBB length did not change beyond sub micron levels when mounted between socket #1 and the spindle. A major portion of this length change was due to the loose jack screws of the machine. After adjustment of the machine jack screws, the LBB length changes were measured again. The change in LBB length for socket #1 remained in the sub micron levels as shown in Figure 5-8 while it was reduced to within 2 microns for sockets #2 and #3 as shown in Figure 5-9 and 5-10.

A similar study was also conducted to measure the change in length of the LBB when mounted in sequence between pairs of base sockets during carriage motion. The results of this study after adjusments of the jack screws are shown in Figure 5-11, 5-12 and 5-13. It can be seen from these plots that the change in LBB length is controlled to within 2 microns for all cases. The rotation matrix between the machine and LBB coordinate frames was also measured for various positions of the carriage motion and was not found to be a function of carriage position. Before drawing any conclusions from these results, it was decided to study the effect of

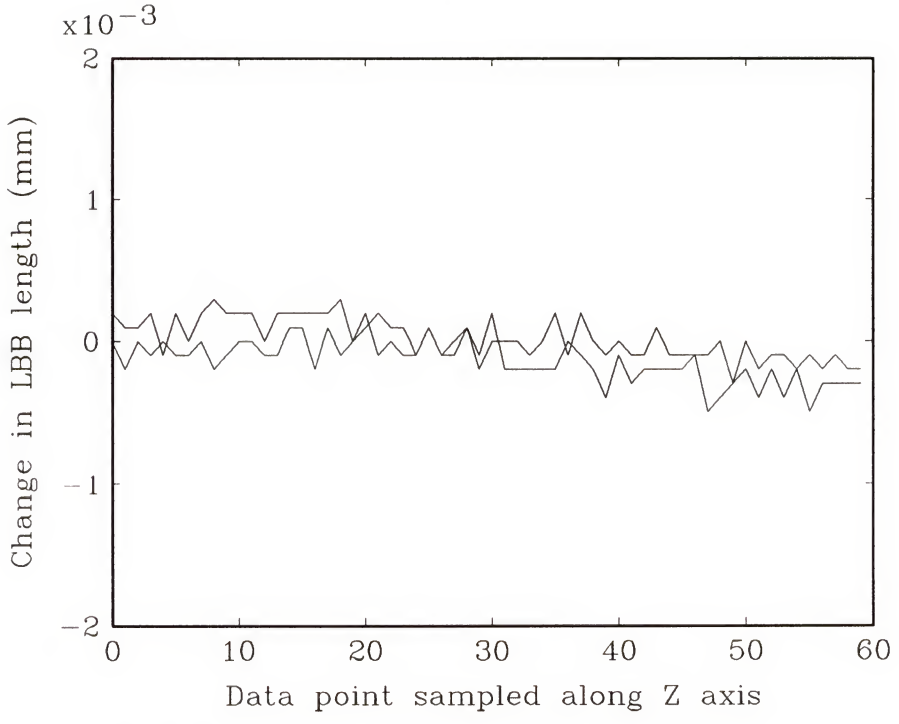


Figure 5-7 LBB length change between base socket #1 and spindle socket.

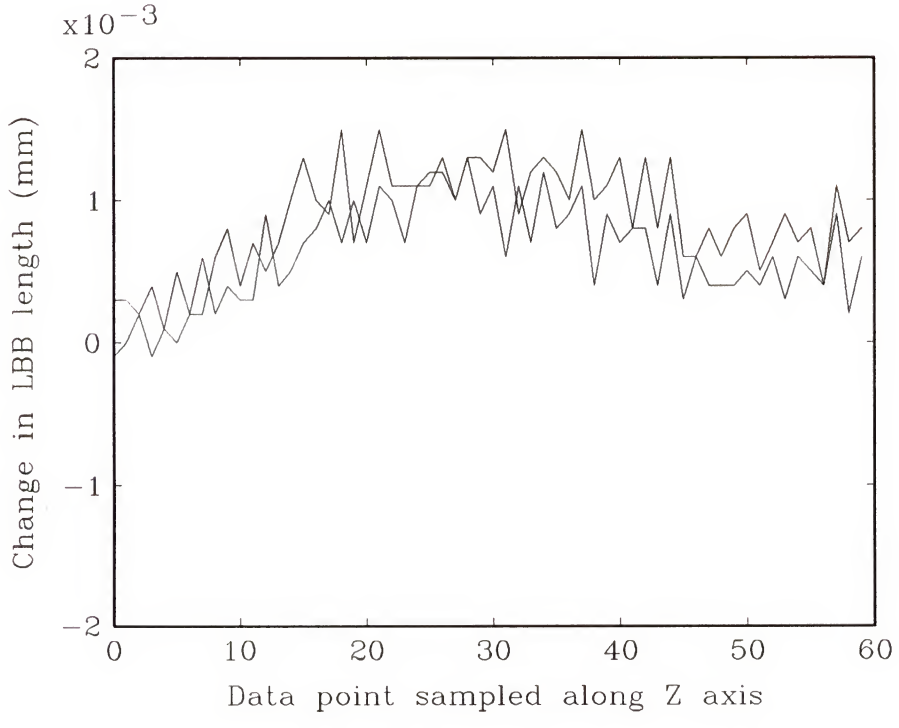


Figure 5-9 LBB length change between base socket #2 and spindle socket.

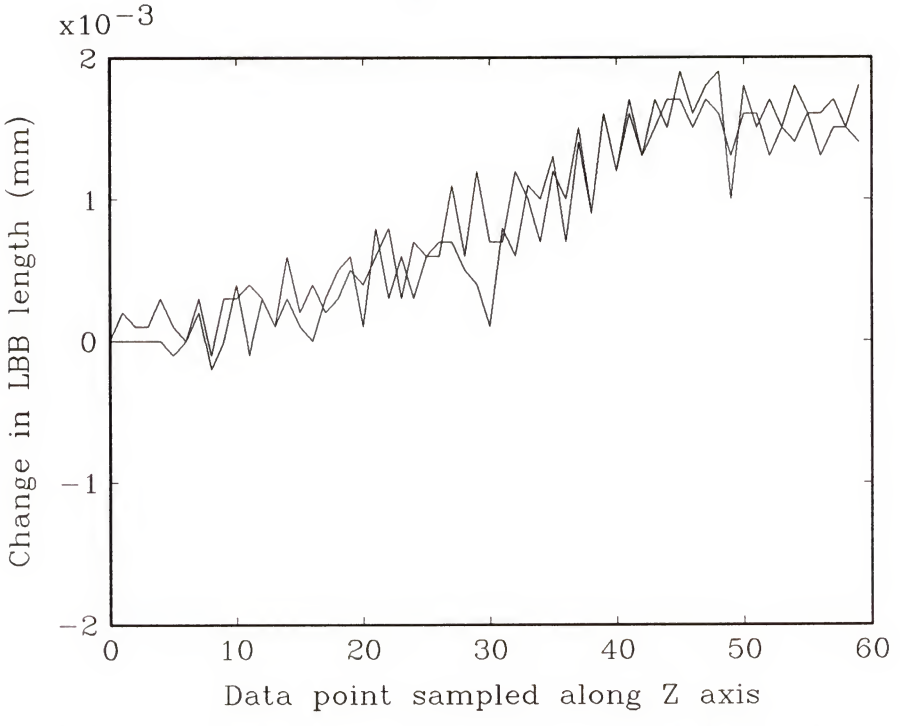


Figure 5-10 LBB length change between base socket #3 and spindle socket.

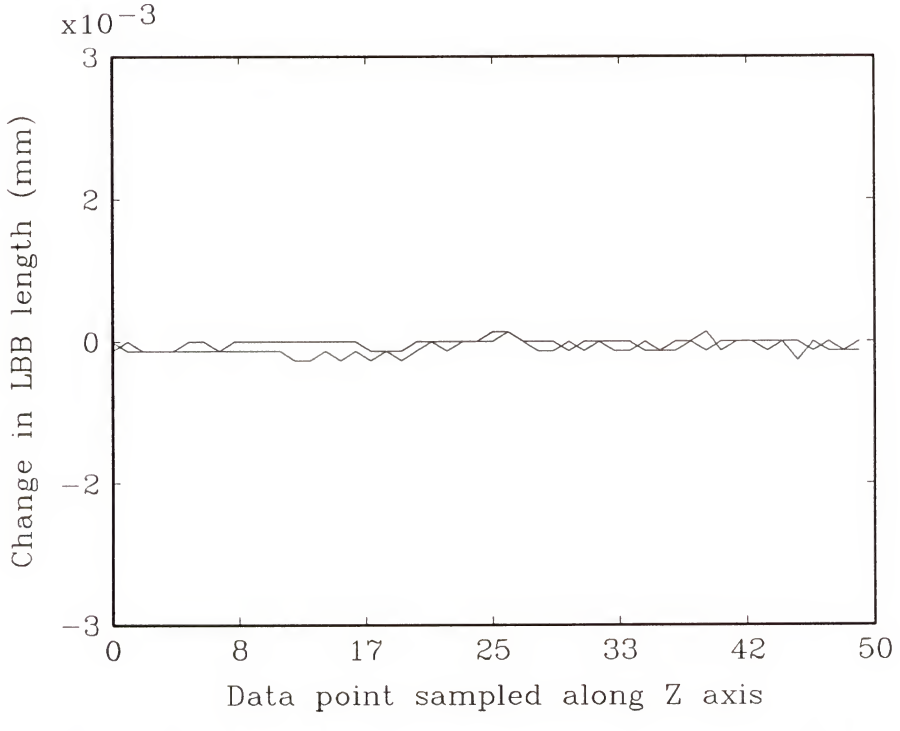


Figure 5-11 LBB length change between base sockets # 1 and #2.

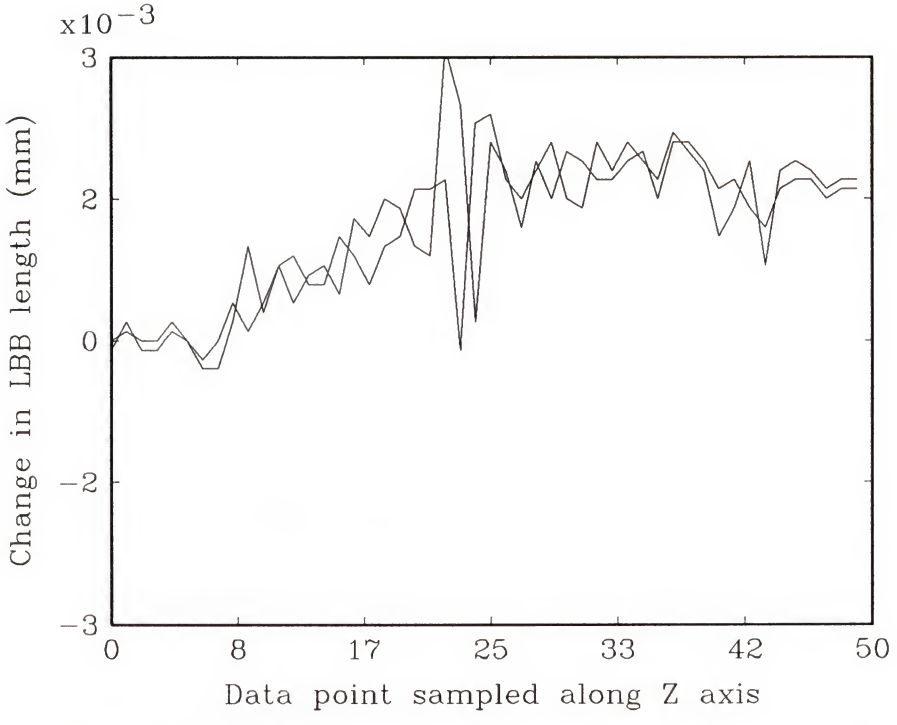


Figure 5-12 LBB length change between base sockets # 2 and #3.

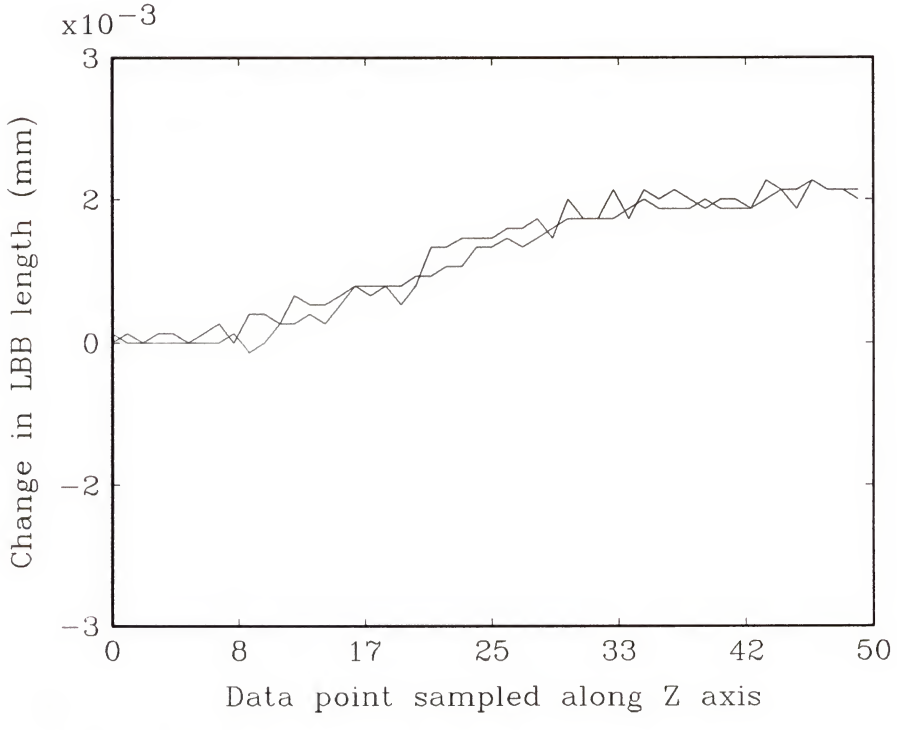


Figure 5-13 LBB length change between base sockets #3 and #1.

carriage motion on the measurement of spindle errors. The measurement of spindle errors using the LBB is outlined in the next section followed by the results of this case study.

LBB Based Measurement of Spindle Thermal Drifts

Spindle thermal drift is caused by temperature gradients introduced into the spindle structure during machining. While thermal drifts of the spindle can be modeled in general as a rigid body with six degrees of freedom (and hence six components of errors), only three components are important to the overall accuracy in the case of a two axis turning center. They are, the axial thermal drift, which is the displacement of the spindle along the carriage (i.e., Z axis), the radial thermal drift, which is the displacement of the spindle along the cross-slide or perpendicular to the carriage motion (i.e., X axis), and the tilt thermal drift, which is the rotation of the spindle axis in the XZ plane of the turning center. These components are depicted in Figure in 5-14.

The conventional method to measure spindle drifts utilizes capacitance probes and a precision ground test artifact [33, 49]. The artifact is mounted on the spindle and a capacitance gage nest is mounted on the turret as shown in Figure 5-15. The a The difference in the readings of probe #1 and probe #2 is used to calculate the tilt in the XZ plane while the average of Probe #1 and Probe #2 readings provide the radial drift. The axial drift is directly measured using Probe #3. The LBB based measurement technique for spindle drifts uses sequential trilateration. The tool

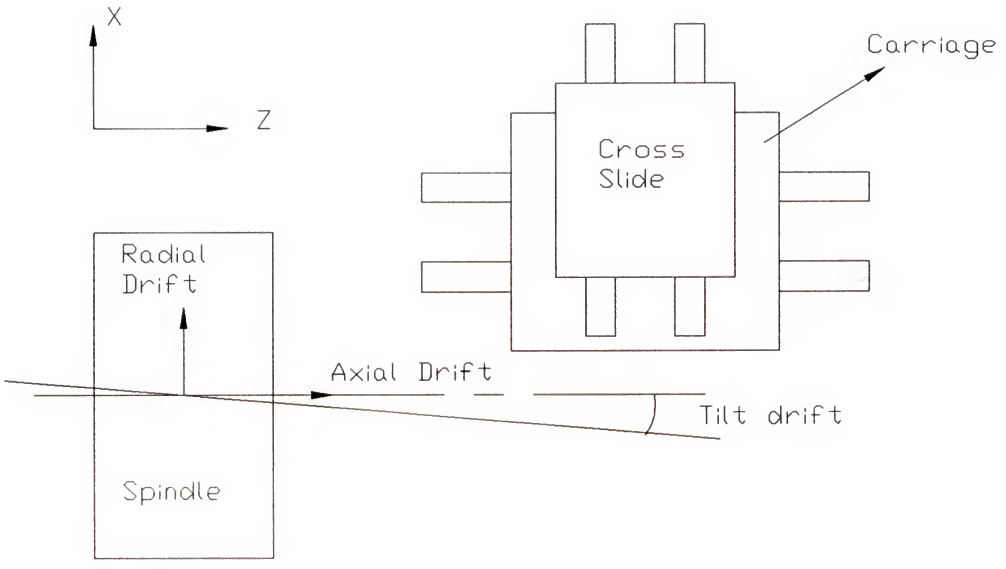


Figure 5-14 Components of spindle thermal drift for a two axis turning center.

socket is replaced by the spindle socket. The LBB is mounted in between the spindle socket and the three base sockets in sequence and the LBB lengths are sampled at evenly spaced angular positions of the spindle for each case while the spindle is rotated at a speed of 1 rpm. This speed was sufficient to measure the LBB lengths to a repeatability of 1 micron. The LBB lengths are transformed into LBB coordinates using equations 5-1 through 5-7 and eventually into machine coordinates using ${}_{B}T^{M}$ matrix. The experimental set up to measure these LBB length between base socket#3 and the spindle socket is shown in 5-16.

The two additional spindle sockets seen in these figures are mounted in order to ensure that the spindle is balanced during rotation at high speeds (required during warming of the spindle). The machine coordinates of the sampled points form a circle in space, which is used to determine the spindle center and spindle axis

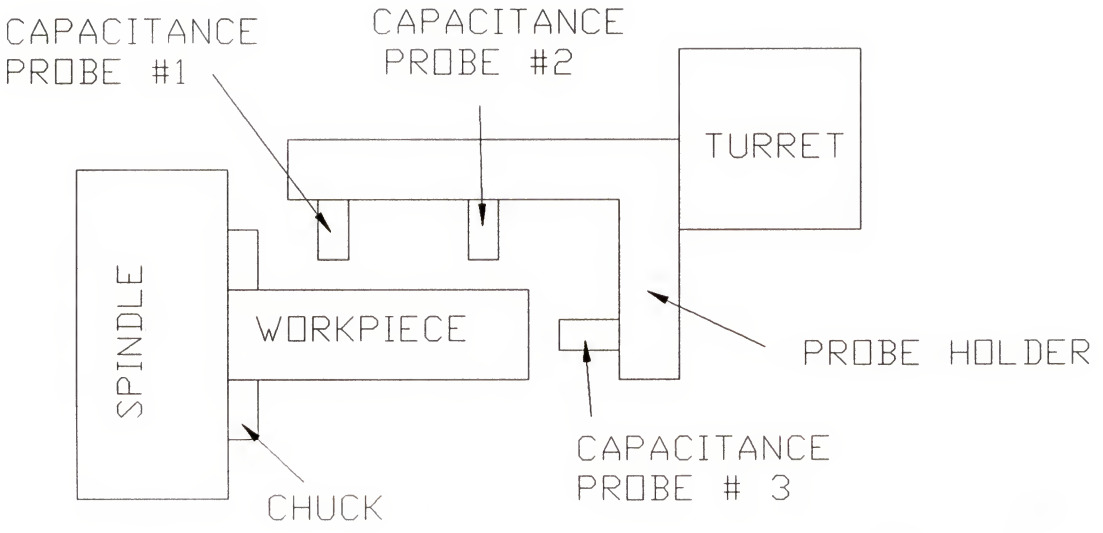


Figure 5-15 Capacitance probe set-up to measure spindle drifts on turning center.

Figure 5-16 LBB measurement between base socket #1 and the spindle socket.

orientation. The axial and radial thermal drifts are measured by computing the difference between the spindle center coordinates at any thermal state from that at the cold state. The tilt thermal drift is measured by computing the difference between the spindle tilt in the cutting plane of the turning center (evaluated from the spindle axis) at any thermal state from that at the cold state.

In order to determine the spindle axis and the spindle center from the sampled data, three different methods were developed and compared. The first method computes the spindle center as the centroid of the sampled points. A set of vectors V_i were defined from the computed spindle center to every sampled data point (x_i, y_i, z_i) as shown in Figure 5-17. The spindle axis 'n' was determined from vectors V_i as follows:

$$n = \frac{\sum_{i=0}^{N-1} V_i \; X \; V_j}{N} \Big|_{normalized} \tag{5-8}$$

where N is the total number of samples and V_j is a vector approximately 90° apart from V_i. The angular separation of 90° was chosen as the cross product of two vectors is most sensitive at this angular separation.

The second method defines the vectors V_i and V_j directly from the sampled data points as shown in Figure 5-18. The spindle axis 'n' is computed from these vectors using equation 5-8. The sampled data points are then rotated into the spindle coordinate frame. This is done by using the spindle axis 'n' to define a transformation ${}_{M}T^{S}$ from the machine coordinate frame to the spindle coordinate

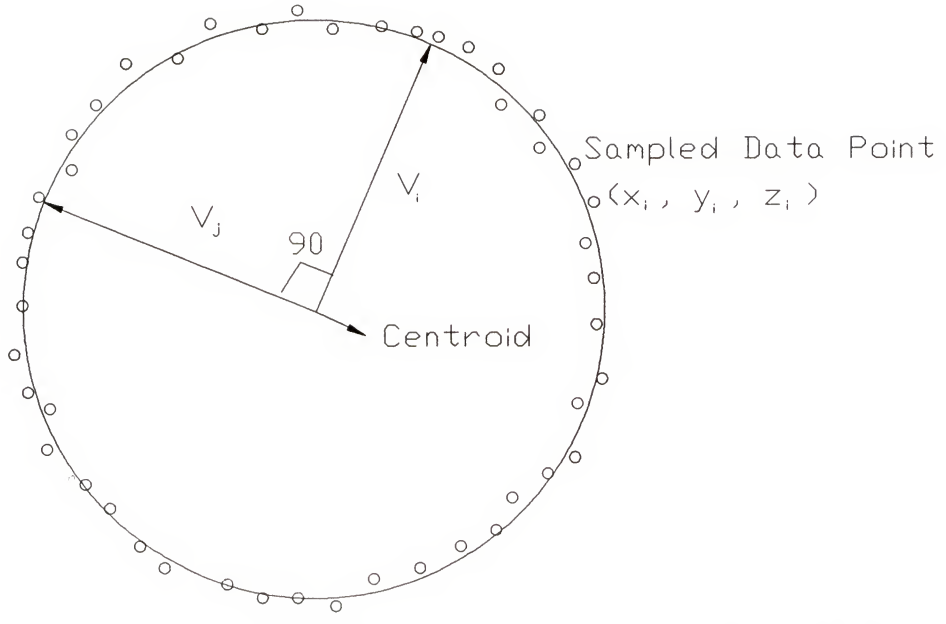


Figure 5-17 Schematic of two chosen vectors V_i and V_j in Method I.

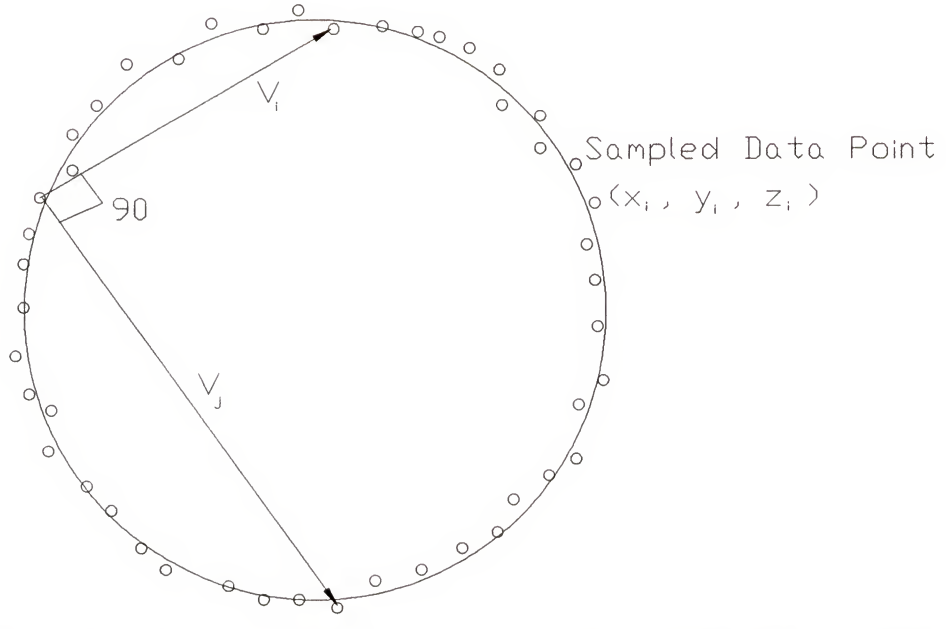


Figure 5-18 Schematic of chosen vectors V_i and V_j for Methods II and III.

frame (using the technique as described in Appendix B). The resulting x and y coordinates lie in the spindle plane (z = constant). For each (x_i, y_i), two other points (x_j, y_j) and (x_k, y_k) sampled approximately at 120° and 240° from (x_i, y_i) are picked. A center is determined from each set of (x_i, y_i) , (x_j, y_j) and (x_k, y_k) using the three point form of the circle [96]. The spindle center is then computed as the average of these centers.

The third method determines the spindle axis in exactly the same manner as the second method. The sampled coordinates are also similarly rotated into the spindle plane using the computed normal. The x and y coordinates are then best fit onto a circle by computing its discrete fourier transform (DFT). This is possible because Hamming [97] proves that for a finite set of data points, the formal fourier coefficients are the coefficients of the least squares fit to the data. The (x_i, y_i) coordinates are formulated into the DFT expression by transforming them into polar coordinates (r_i, θ_i) (for i = 0, . . ., N-1). The centroid of (x_i, y_i) is taken as the first guess center. The zeroth and first harmonics of the DFT are computed as follows:

$$DFT(n) = \frac{\sum_{i=0}^{N-1} r_i e^{j(\frac{2\pi}{N})in}}{N} \qquad (n = 0, 1) \qquad (5\text{-}9)$$

The zeroth harmonic DFT(0) provides the best fit radius to the data. The real and imaginary parts of the first harmonic DFT(1) provides the eccentricity in the X and Y directions of the first guess center (i.e., the centroid) to the best fit center. The

higher harmonics provide a measure of the residual errors.

In order to compare the three methods a case study was conducted on the two axis turning center . The deviation in the estimated normals at each point for three methods were computed. Methods II and III have smaller deviations in the estimated normals in comparison Method I. The radius of all data points (x_i, y_i) in the spindle plane were computed using the center derived by each method. The results are plotted in r vs. θ form as shown in Figure 5-19. If the derived center is the best fit center, there should be a minimal deviation in the radius (the deviations being the residual errors) and the r vs. θ plot should be nonperiodic (thus eliminating the possibility of the estimated center from having any eccentricity). From Figure 5-19, it can be seen that the r vs. θ plot for method I has a periodic waveform with

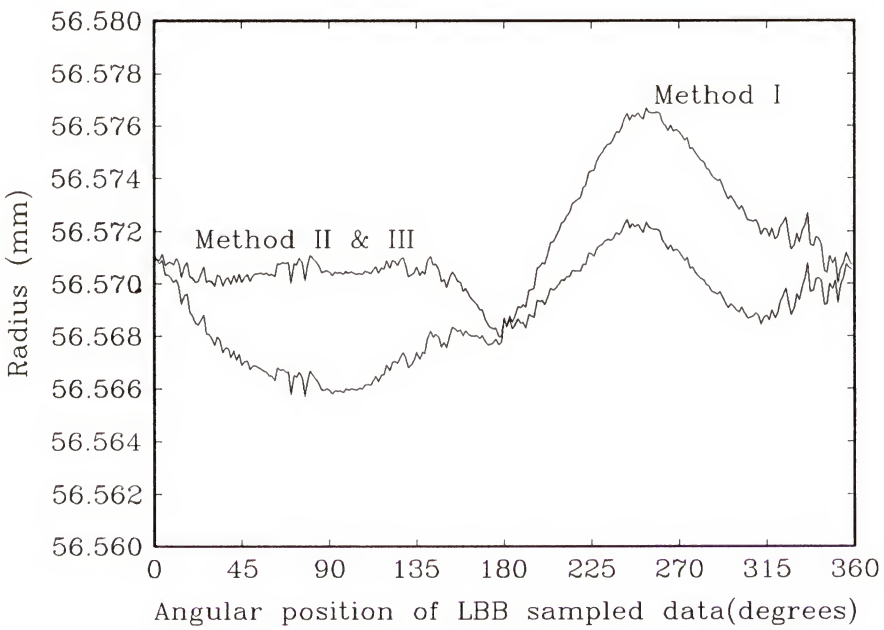


Figure 5-19 Radius versus angular position comparison for the three methods.

a maximum deviation in radius of about 10 microns. Methods II and III estimate the same center and the r vs. θ plot for these two cases (superimposed on each other) have a non-periodic output with a maximum deviation in radius of 3 microns.

Method I is more inaccurate than the other two methods because it depends on the centroid for all its computations and the centroid in general provides only an approximate estimate of the true center. While methods II and III provide identical results, method II is more robust than method III. This is because, method III requires the data to be equally spaced in θ in order to converge to the correct center in a single iteration. For non-equally spaced data in θ, method III converges to the correct center by iterating the DFT computations using the center estimated from the previous DFT until there is no further improvement. There are no such restrictions in the case of method II. In this research, method II will be adopted to derive the spindle axis and center.

As mentioned in the previous section, a case study was conducted to evaluate the effects of deflections of machine (during carriage motion) on the measurement of spindle center and the spindle axis. The carriage was moved to different positions along the Z axis and the spindle center and axis was determined using Method II. The results of this case study is presented in the form of Table 5-1. These results were obtained after adjusting the machine jack screws as described in the previous section. It can be seen that the maximum deviation in the spindle center coordinates are 3 microns in X and 2 microns in Z coordinate. The maximum deviation in tilt is 0.0005°. From our previous knowledge of the range of X and Z positional errors

as well as spindle drifts (based on ASME B5.54 standard) for the MAZAK turning center, it is known that the thermally induced errors grows to about 40 microns in X axis and 80 microns in Z axis after 6 hours of warming up the machine. Similarly the spindle centers drifted to the order of 35 microns when run at 2000 rpm for 6 hours. The deviations in the spindle center and tilt is minimal compared to the thermal induced growth in these errors Thus the sockets mounted on the I beam and the guideway of the tailstock was considered sufficiently rigid for measuring thermally induced errors of the two axis center.

Table 5-1 Results of case study on effect of machine structure deflections due to carriage motion on spindle center and tilt calculations.

Carriage Position (mm)	Spindle Center X (mm)	Spindle Center Y (mm)	Spindle Tilt (deg)
0	97.932	351.353	0.0065
50	97.933	351.352	0.0063
100	97.934	351.353	0.0060
150	97.935	351.354	0.0060

CHAPTER 6
EXPERIMENTAL RESULTS

The previous chapter explained the methodology to measure the positional error map and spindle thermal drifts using the LBB. In this chapter, the experimental results obtained using this methodology will be discussed. First, a set of initial steps have to be followed before data collection can proceed. These steps include the tool eye setting of the LBB and the measurement of the rotation matrix between the LBB and machine tool frame. A brief justification for choosing the locations of the thermocouples is also provided. This is finally followed by an in-depth discussion of the experimental procedure and results obtained.

Tool Eye Setting for the LBB

In a turning center, all references to the workpiece are made with respect to the spindle axis since the workpiece is machined while being clamped on the spindle. This is unlike a machining center where the workpiece is mounted on a work table and the reference coordinate frame can be set to the convenience of the user. In order to locate the cutting tool with respect to the spindle in the machine coordinates, the turning center is provided with a tool eye station as shown in Figure 6-1. This station has four touch sensors (two for each axis) and an LED to indicate

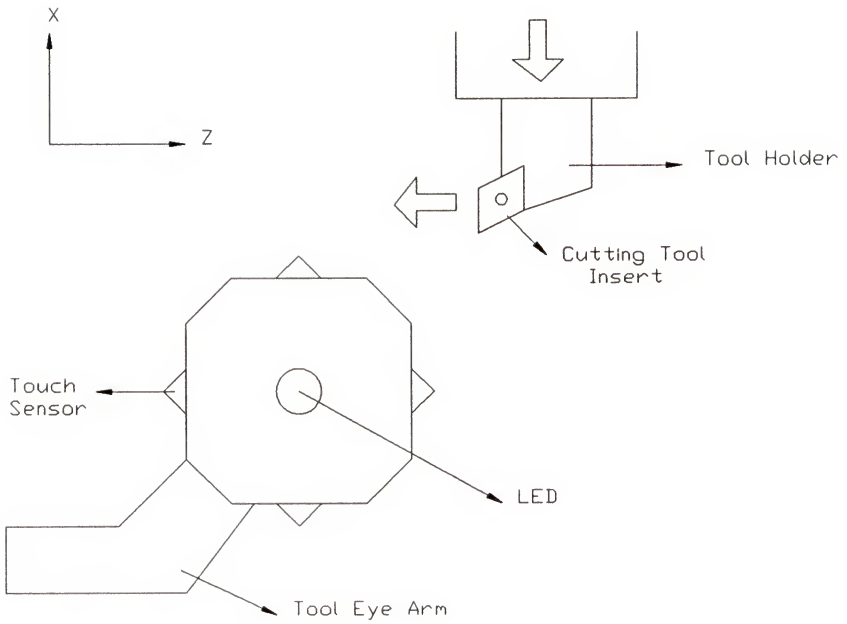


Figure 6-1 Tool Eye setting for the cutting tool of the turning center.

if a contact has been established with the touch sensor. In the normal mode (i.e., when no contact has been established) the LED is on. The cutting tool is fed (in the jog mode) to make the tool tip contact the X axis sensor. Axis feed is automatically stopped when the tool makes contact with the sensor through feedback from the LED which acts as a switch. The procedure is repeated for the Z axis. In order to collect useful machine tool error data, the LBB must measure all its measured errors with respect to the spindle axis. This process begins by mounting the LBB between the base socket #3 and the tool socket. The precision sphere of the LBB attached to the tool socket is first located with respect to the Z sensor of the tool eye station as shown in Figure 6-2. This process is repeated for the X sensor of the tool eye station. These steps help only in locating the sphere surface in machine coordinates.

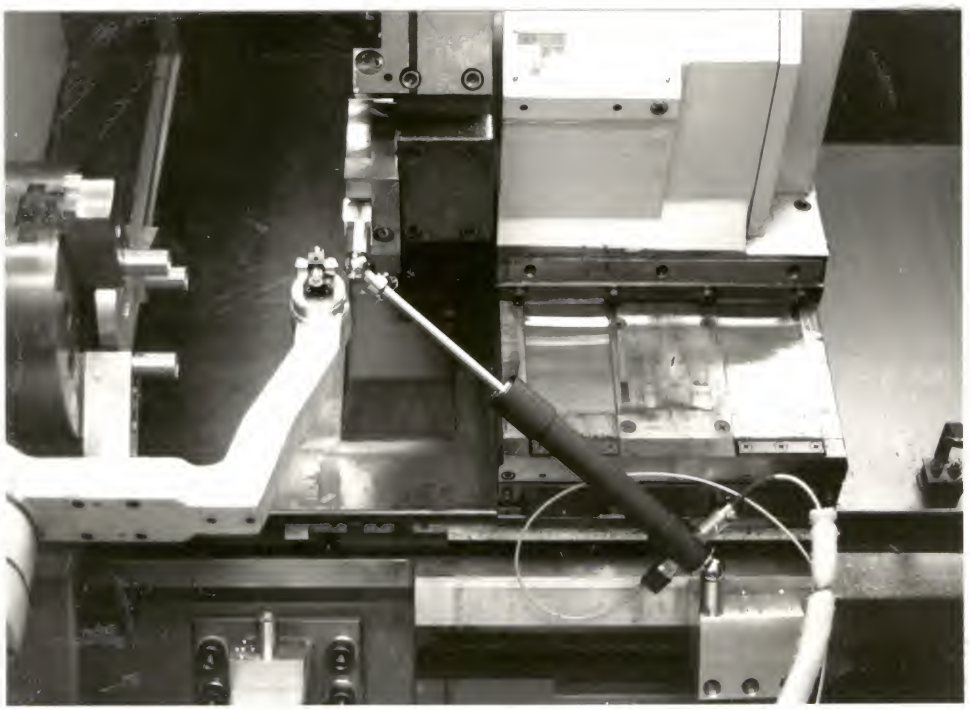


Figure 6-2 Locating a precision sphere of LBB using the tool eye station.

However, LBB lengths measured during the measurement of positional errors are between the centers of the two precision spheres. Hence it is important to locate the center of the sphere. In order to locate the center of sphere in machine coordinates, the radius of the sphere is subtracted from the X and Z machine coordinates of the sphere surface as measured by the tool eye. The radius of spheres mounted on the LBB was measured on the M60 coordinate measuring machine at the Oak Ridge National Laboratory and the diameter of the sphere attached to the tool socket was measured to be 1.00028 inches. This data was used to locate the center of the precision sphere of the LBB in machine coordinates and thus ensure that the LBB data can be directly used for error compensation.

Measurement of Rotation Matrix

The coordinates derived from the LBB are with respect to the LBB coordinate system. These coordinates are converted into machine coordinates by using the transformation matrix ${}_{B}T^{M}$ (as shown in Appendix A). Each axis is fitted by a straight line (using a least squares method) with a set of ten points taken at equal intervals of one inch. Thermally induced error motions cause the machine frame to rotate and translate from its cold state location. The translations are taken into account by normalizing the errors measured with respect to the origin of the machine frame at cold state. In the same manner, the rotations of the machine frame are taken into account by transforming the ball bar coordinates at any thermal state into the machine coordinates using the rotation matrix at the cold state. Thus the rotation matrix needs to be evaluated only at the cold state of the machine. A typical rotation matrix measured between the LBB and machine frame at the cold state is shown below.

$$R_{cold} = \begin{bmatrix} -0.788484 & 0.110009 & 0.605137 \\ 0.614884 & 0.117782 & 0.779772 \\ 0.014508 & 0.986927 & -0.160512 \end{bmatrix} \quad (6\text{-}1)$$

The first row represents the unit vector of the X axis of the machine frame in LBB coordinates. Similarly, the second and third rows represent the unit vectors of the Y and Z axis in LBB coordinates respectively. For example, it can be seen from row 3 (or column 2) of R_{cold} that the Y axis of the LBB frame is approximately parallel to the Z axis of the machine frame.

Location of Thermocouples

Before going into the experimental details of measuring thermally induced errors in the turning center, it is necessary to briefly discuss the possible heat sources to be considered. This problem has been researched by Tlusty and Mutch [17]. According to [17], it is possible to estimate the internal heat sources in a machine tool fairly well based on engineering judgement. In this research, we identify our heat sources and thermocouple locations based on the guidelines provided in [17].

First, the carriage and cross-slide of the turning center are considered. The heat generated by the sliding of the guideways can increase with continuous motion of the carriage and cross-slide. Hence, four thermocouples T_1, T_2, T_3, and T_4 were mounted on the guideways of the carriage and cross-slide as shown in Figure 6-3. The position feedback for the axes is provided by the encoders attached to one end of the leadscrew. Rapid traverse of the carriage and cross-slide can generate significant heat between the leadscrew and the nut. Hence, thermocouple T_5 (as shown in Figure 6-3) was mounted on the leadscrew bearing housing of the carriage. Similarly, thermocouples T_6 was mounted on the leadscrew bearing housing of the cross-slide. The temperature change of ambient air is monitored as a separate heat source using thermocouple T_7. This was necessary because the room temperature in the laboratory is not adequately controlled. The other main internal heat source in the turning center is the heat generated in the spindle bearings. This heat is monitored via thermocouples T_8 through T_{13} mounted on the spindle bearing housing as shown in Figure 6-4. The difference (T_{13} - T_8) provides a measure of the thermal

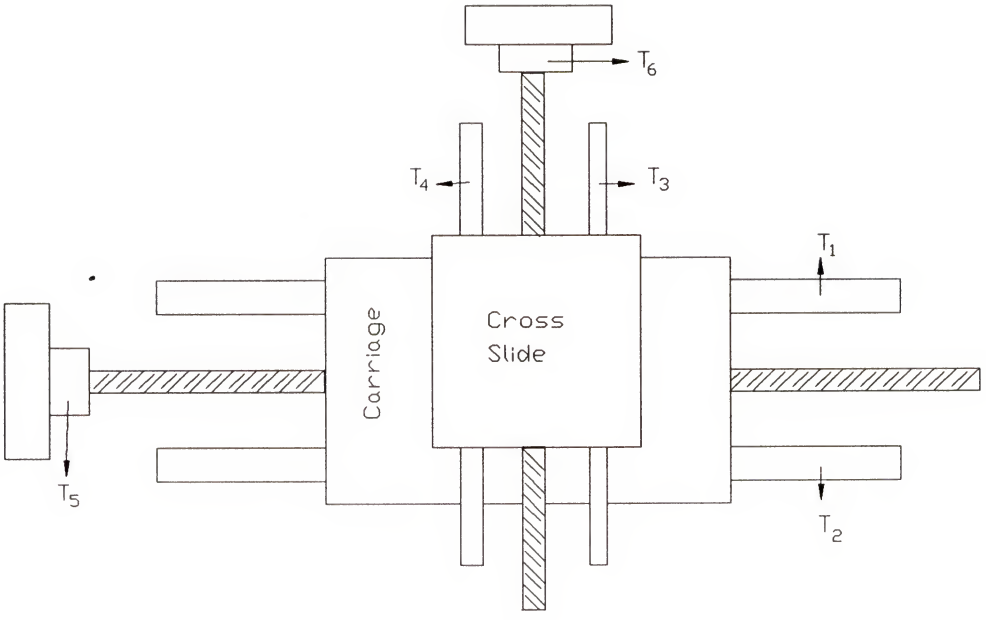


Figure 6-3 Location of thermocouples on carriage and cross-slide elements.

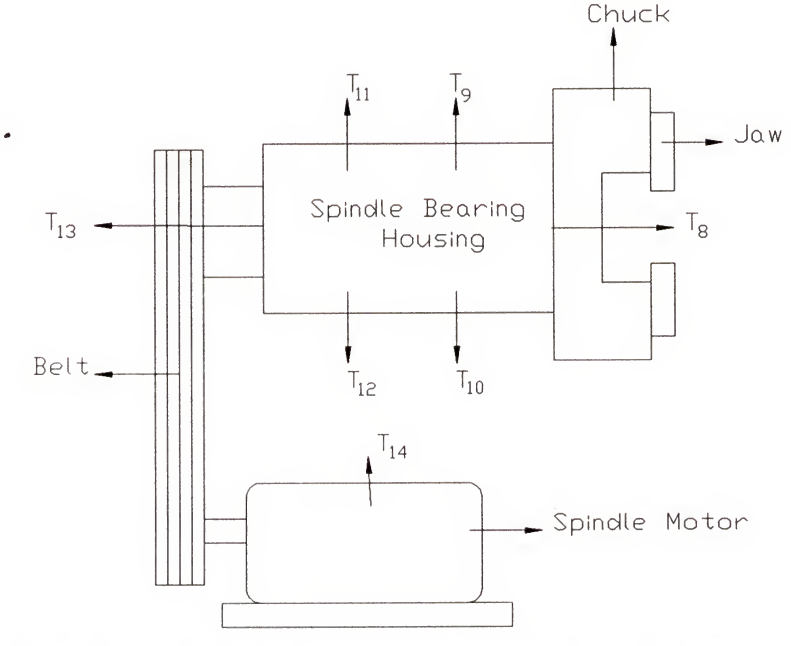


Figure 6-4 Location of thermocouples on the spindle elements.

gradients in the Z (axial direction). These gradients occur as a result of more heat generated at the rear of the spindle bearing compared to the front bearing. This is because the rear part of the spindle bearing housing is located in close proximity tothe belt drive as shown in Figure 6-4. Similarly, (T_9 - T_{10}) and (T_{11} - T_{12}) provide a measure of the thermal gradients in the X (radial) direction. In addition to the spindle bearing, the heat generated by the spindle drive motor is monitored using thermocouple T_{14}. The heat generated by the motor can be transferred to the other components of the machine by radiation, convection, and conduction.

The heat generated during the cutting process is another important heat source. As explained in chapter 3, McClure [16] has shown in his work at University of California at Livermore that about 80% of the heat input to the workpiece during the cutting process can be removed using a coolant. The chips generated during the cutting process can become a significant heat source if the they are left to accumulate on the machine. The coolant helps in quickly removing the chips from the workpiece and they normally pile up in the coolant tank along with the coolant. This tends to increase the temperature of the coolant. Hence the coolant is required to be temperature-controlled. While most of the currently available machine tools in industry are equipped with a temperature control unit for the coolant bath, the turning center used for this research is not. There are two options to overcome this problem. The first option is to perform rough machining and then perform the finish machining in a separate setting at a much later time. Since the heat generated during finish machining is small, this method minimizes the effects of heat generated during

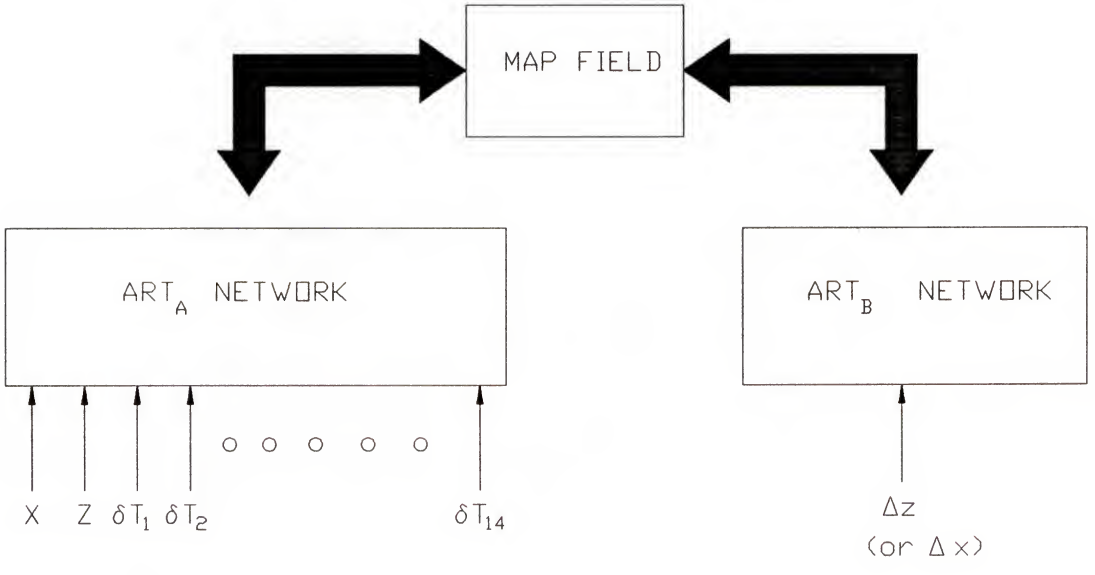
the cutting process. The second option is to minimize the cutting time by minimizing the stock to be removed by the roughing operation. The roughing and finish machining can then be performed with coolant at the same setting. This option will not heat up the coolant significantly as the rough machining time is minimized. A drawback of the second option is that it is not possible to machine more than a few parts because the coolant (which absorbs the heat generated during the cutting process) begins to heat up as it gets used more frequently. Our goal is to demonstrate that real-time geometric and thermally induced error compensation is achievable using the LBB and Fuzzy ARTMAP neural network combination. In order to demonstrate this goal, it is sufficient to machine just a few parts at different thermal states. Hence it was decided to adopt the second option in this research. As a precaution, the coolant tank was covered with a net (or sieve) in order to prevent the hot chips from falling into the coolant tank.

Results

As mentioned earlier, the experiment begins with a tool eye setting of the LBB. This is followed by a self-initialization of the LBB using the three socket technique. The LBB lengths between the three base sockets are measured. An illustration of the base length measurement between socket#2 and #3 is provided in Figure 6-5. The rotation matrix is then measured at the cold state. The Fuzzy ARTMAP neural network is to be trained on the data measured by the LBB for a standard thermal duty cycle. The input vector to the ART_A module consists of

temperature changes at 14 different locations on the machine from its cold state and the nominal machine coordinates of the tool socket in the complement coded form. These temperatures were measured using the thermocouples at the following locations of the machine: the slide ways, leadscrew nut, the ballscrew bearing housing for each axis, spindle bearing housing temperatures (as mentioned in the previous section), the spindle drive motor and ambient air. The inputs to ART_A and ART_B modules of the Fuzzy ARTMAP are shown in Figure 6-6. It should be noted that two thermocouples were added for each heat source. The redundancy is deliberately introduced in order to provide the neural network a certain amount of fault tolerance. In other words, the network is provided with data from an extra thermocouple per heat source so that it can function effectively during compensation

Figure 6-5 Base length measurement between socket #2 and #3.



x - cross slide position in machine coordinates

z - carriage position in machine coordinates

δT_1 - temperature of ith heat source from its cold state

Δz - LBB measured total z positional error at cutting tool

Δx - LBB measured total x positional error at cutting tool

Figure 6-6 Inputs and outputs to the Fuzzy ARTMAP neural network.

even if one of the thermocouples came loose. The total positioning error at the cutting tool (including the spindle drifts) measured by the LBB is fed as input to the ART_B network in the complement coded form. The temperature measurements are provided to the network as changes from cold state in order to capture the thermal gradients in the various heat sources. These gradient patterns input to the network will be different for heating and cooling as the time constants for heating and cooling are different [17].

The Fuzzy ARTMAP attempts to correlate the temperature distributions of the various heat sources to the error of the machine (as measured by the LBB). The approach of correlating machine tool errors to temperature distributions was demonstrated by Donmez [33] and Chen [49]. The basic premise in this approach is that it is possible to predict the temperature distribution at any given time by monitoring the temperatures at a few given locations This is attributed to the fact that there are only a few major heat sources in the machine tool (like the leadscrew and spindle bearing) and their temperature distributions are coupled with each other. Furthermore they showed that the error behavior of the machine is unique with respect to this temperature distribution.

The Fuzzy ARTMAP is trained on data collected during a thermal duty cycle (TDC1). It consists of a series of measurement periods with a warming period in between each of them until temperature changes of the various heat sources stabilize (or reach thermal equilibrium). The measurement period consists of measuring the total positioning errors in X and Z directions over a grid of sixty four (8 X 8) points

using the LBB. These grid points are located at equal intervals of 25.4mm (1 inch) in Z and 12.7mm (0.5 inches) in X direction. These measurements are conducted by mounting the LBB between one of three base sockets and the tool socket. A CNC program is used to locate the LBB at the nominal X and Z coordinates of each grid. This process is repeated for the other two base sockets. An example of one such measurement is illustrated in Figure 6-7. In addition to measuring the total positional error map, spindle drifts are also measured during each measurement period. The LBB is mounted between each base socket and the spindle socket. The spindle center and tilt of the spindle axis in the XZ plane are obtained by measuring the machine coordinates of a spindle socket from LBB data using Method II as explained in the previous chapter. During this measurement, the spindle was run at

Figure 6-7 LBB length measurement between base socket #2 and the tool socket.

1 rpm and the three LBB lengths were sampled at 256 points (i.e., a point for every 1.40625°) on a circle traced by the spindle socket. In order to ensure that the three lengths are sampled at the same angular position of the spindle, the computer was interfaced to the spindle encoder of the turning center. This encoder had 1024 lines and hence the minimum resolution to which the angular position of the spindle could be resolved was 0.0878906° (accounting for the quadrature form of the pulse train). However, the LBB data could not be measured at this resolution owing to a limit on the computational speed in simultaneously sampling the laser and the encoder.

The spindle thermal drifts are measured as the difference between the spindle center at any thermal state compared to that at the cold state. The thermal tilt drift is measured in a similar fashion by computing the difference between the spindle tilt in the XZ plane at any thermal state to that at the cold state. In order to evaluate the LBB method of nominal spindle drifts, a case study was conducted. The axial and radial drifts of the spindle were measured with the spindle running at 1000 and 2000 rpm using capacitance probes (as explained in the previous chapter). The LBB was also used to measure spindle drifts for the same spindle speeds. The results of the two methods were compared as shown in Figure 6-8 through 6-11. The radial tilt drift for TDC1 as measured by the LBB is shown in Figure 6-12. These results show that the LBB is capable of measuring spindle drifts in good comparison to the currently accepted capacitance probe method. The measurement time for spindle drifts was 4 minutes and the measurement time taken for the entire measurement cycle was 12 minutes. It should be noted that before each measurement cycle, the

LBB is initialized and the base lengths are re-measured to account for thermal growth (if any) of the LBB components and the base socket fixtures.

During the warming period, the carriage and cross-slide are exercised to move along a diagonal to their respective ranges of travel (800mm for the carriage and 192mm for the cross-slide) at a constant feed rate of 4000mm/min. In addition, the spindle was run at a constant speed of 2000 rpm. The duration of each warming period is set to 20 minutes. The turning center requires about 6½ hours for the temperatures around the machine tool to reach thermal equilibrium. The machine was then shut down for cooling with a series of measurement cycles during this period. The cooling period lasted for about 6 hours. The data collected during TDC1 simulates machining of large workpieces with several hours of machining and setup time. The temperature profiles of the various heat sources during TDC1 are shown in Figures 6-13 through 6-16.

The inputs to the ART_B module is the total positioning error along the X and Z axis including the spindle drifts and tilts. The spindle drifts and tilts were included into the positional error map of the machine tool to obtain total positional error Δx_i and Δz_i as:

$$\Delta x_i = \Delta x_i^{LBB} - \Delta Z_i \epsilon_y(s) - \delta_x(s) \tag{6-2}$$

$$\Delta z_i = \Delta z_i^{LBB} + \Delta Z_i \epsilon_y(s)^2 - \delta_z(s) \tag{6-3}$$

where Δx_i^{LBB} and Δz_i^{LBB} are the total x and z positional errors measured at grid point

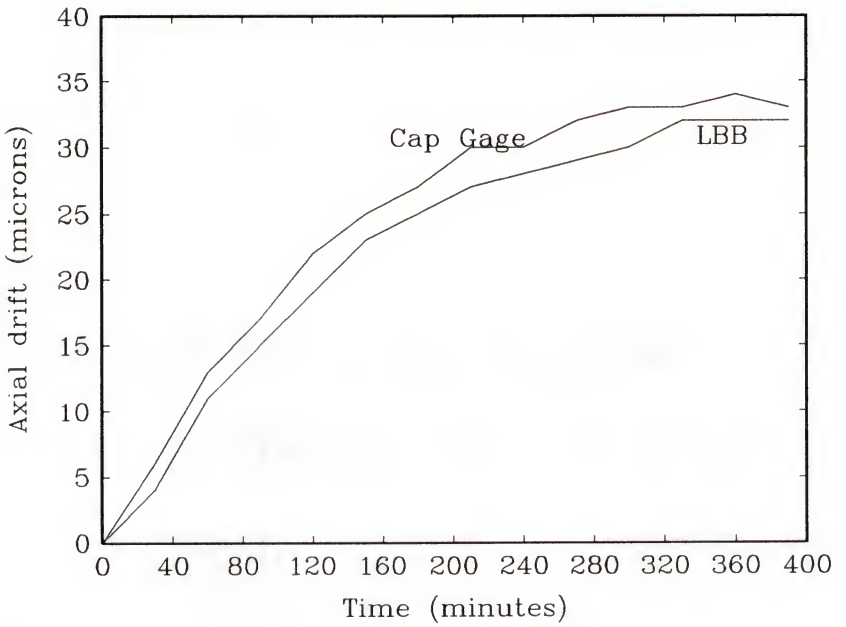


Figure 6-8 Axial drift comparison between LBB and capacitance gage for a spindle speed of 2000 rpm.

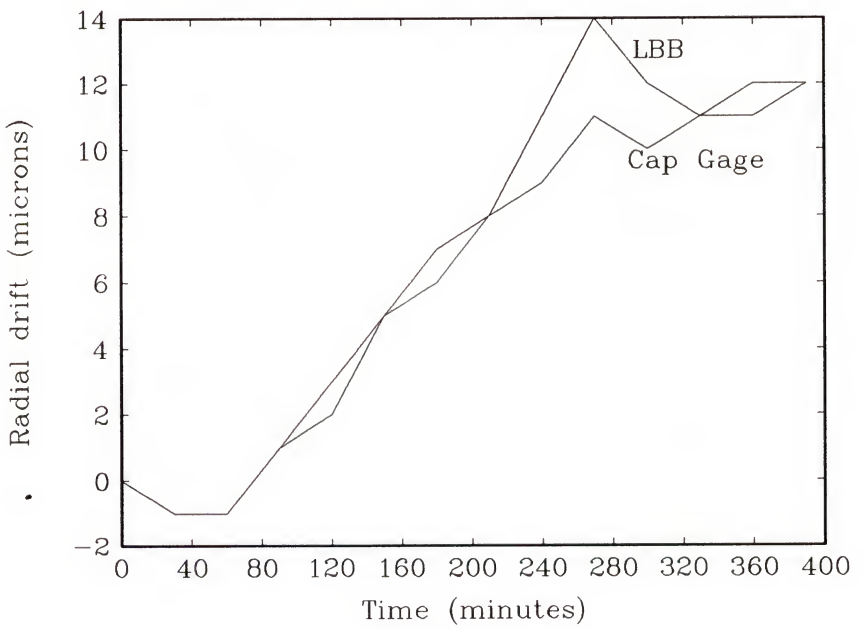


Figure 6-9 Radial drift comparison between LBB and capacitance gage at a spindle speed of 2000rpm.

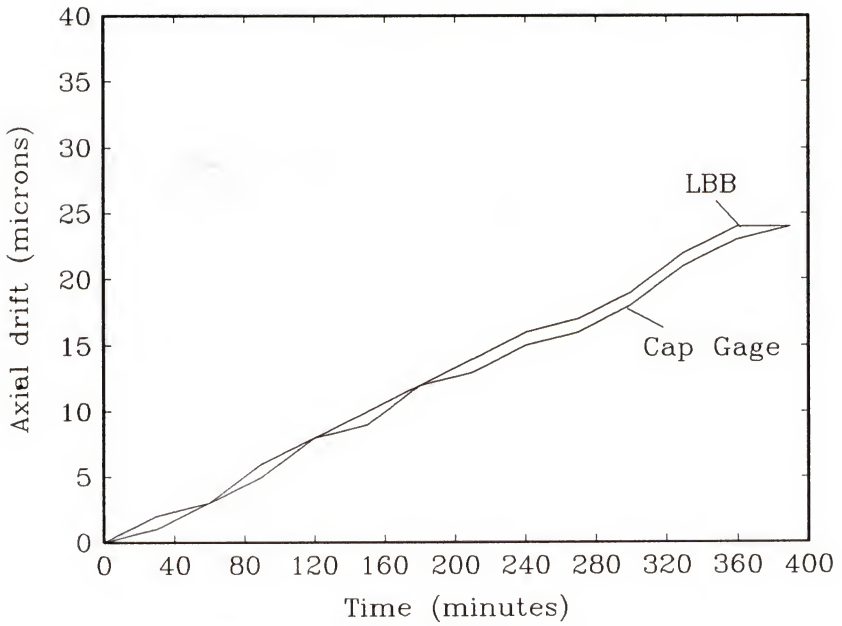


Figure 6-10 Axial drift comparison between LBB and capacitance gage for a spindle speed of 1000rpm.

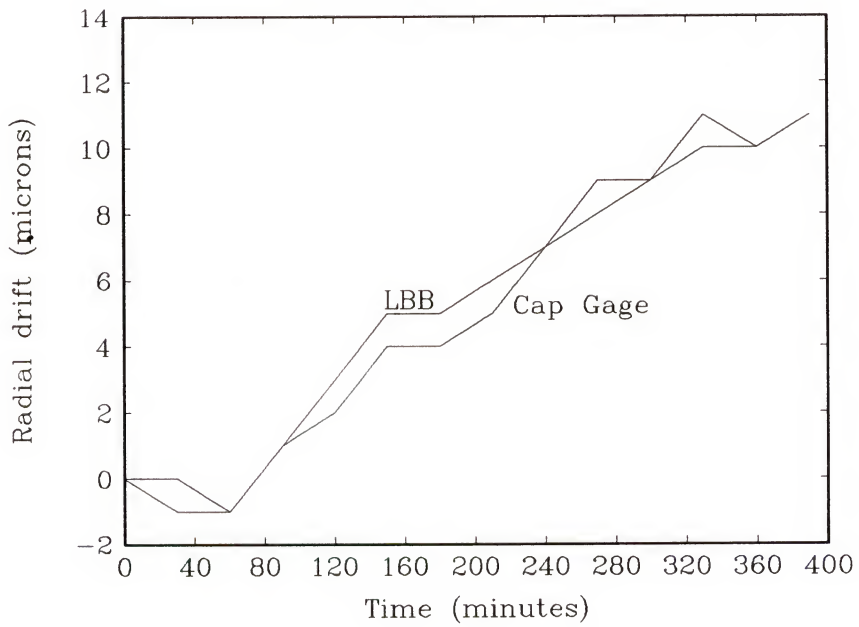


Figure 6-11 Radial drift comparison between LBB and capacitance gage for a spindle speed of 1000rpm.

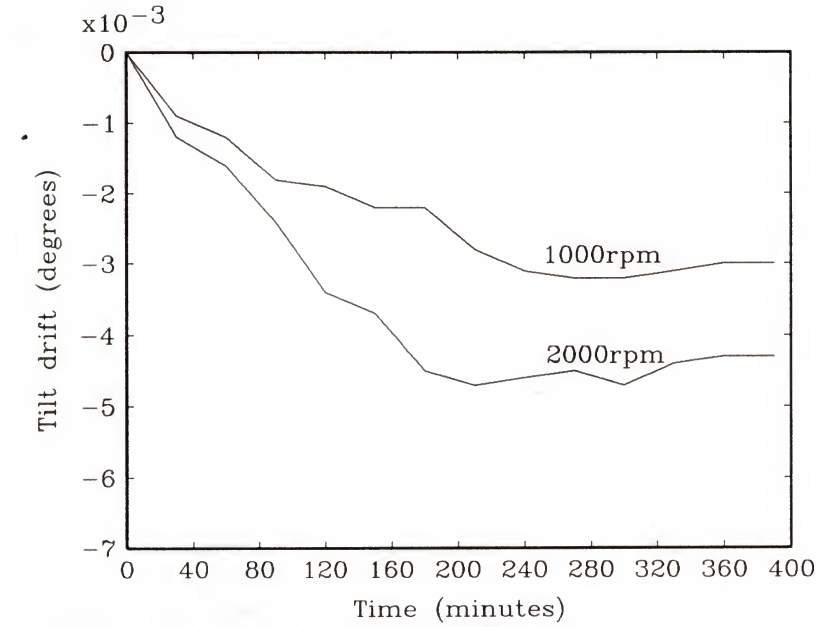


Figure 6-12 Tilt Drift measured by the LBB for spindle speeds of 1000 and 2000rpm.

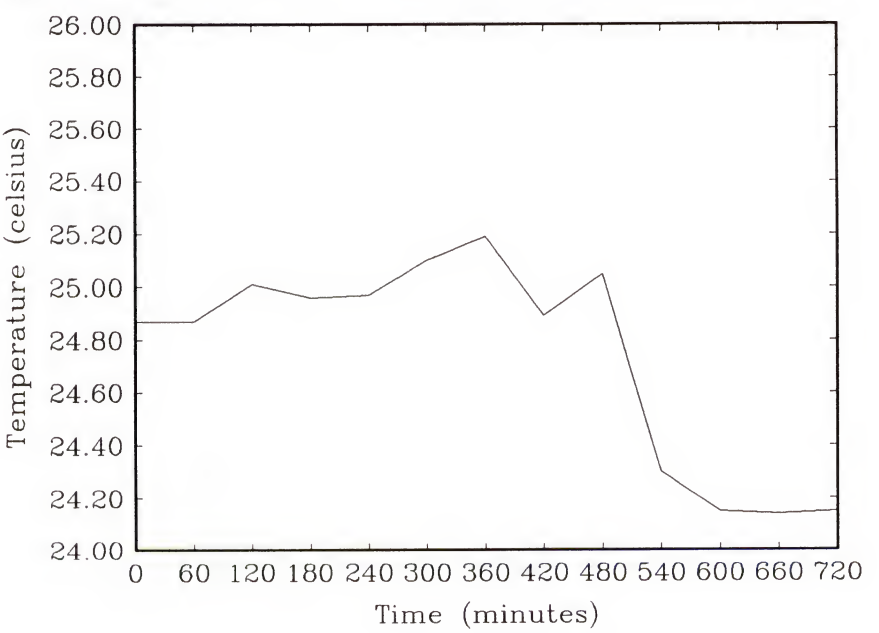


Figure 6-13 Room temperature variations during TDC1.

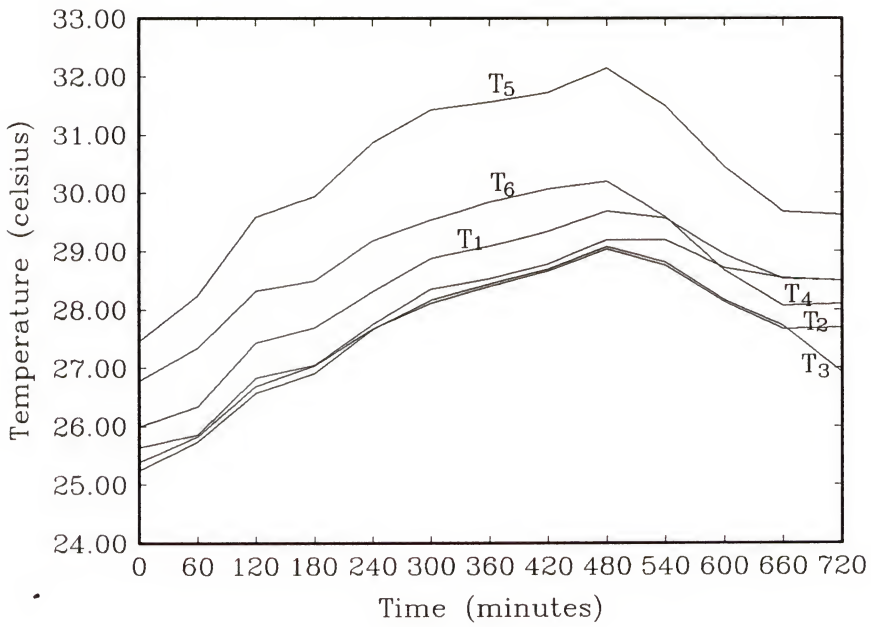


Figure 6-14 Temperature profiles of carriage and cross-slide elements.

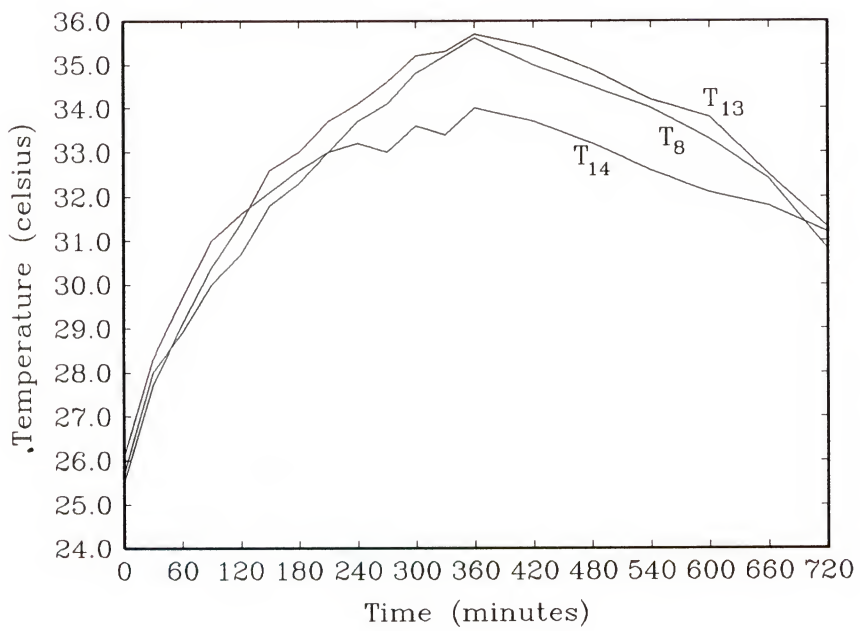


Figure 6-15 Temperature profiles of spindle elements T_8, T_{13} and T_{14}.

#i, $\epsilon_y(s)$ is the tilt of the spindle axis in the XZ plane (in microns/mm) and $\delta_x(s)$ and $\delta_z(s)$ are the axial and radial drifts of the spindle. The physical meaning of equations 6-2 and 6-3 can be seen in Figure 6-17. The middle term of equation 6-3 is a second order term and will be neglected in all our calculations. It should be noted that the spindle drifts measured at spindle speeds of 1000 and 2000rpm were used for training the Fuzzy ARTMAP neural network on TDC1.

The training was performed on-line as follows. Data collected during each measurement cycle using the LBB is converted into a total positional error map using equations 6-2 and 6-3. During each warming cycle, the error map data from the last measurement cycle is used as the training set to train the network. In this manner,

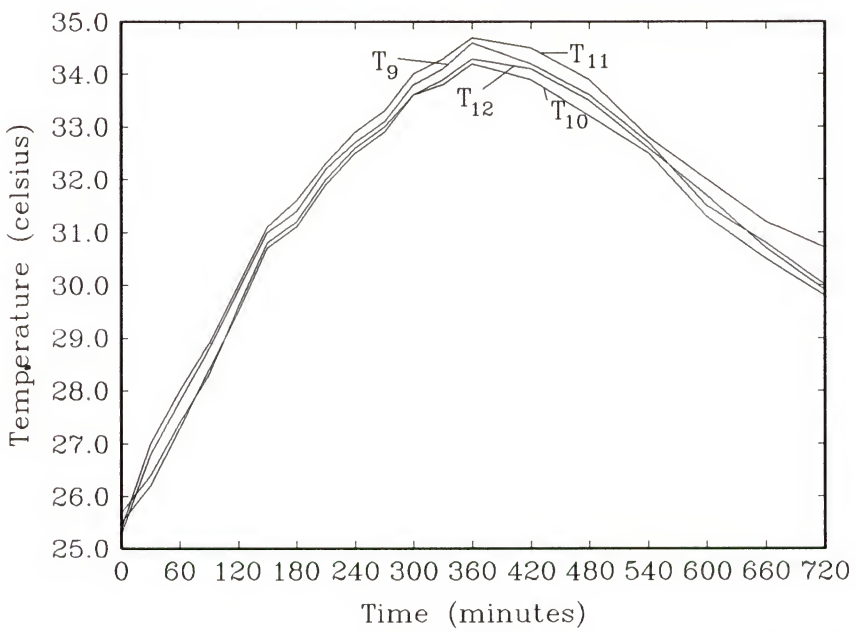


Figure 6-16 Temperature profiles of spindle elements T_9, T_{10}, T_{11} and T_{12}.

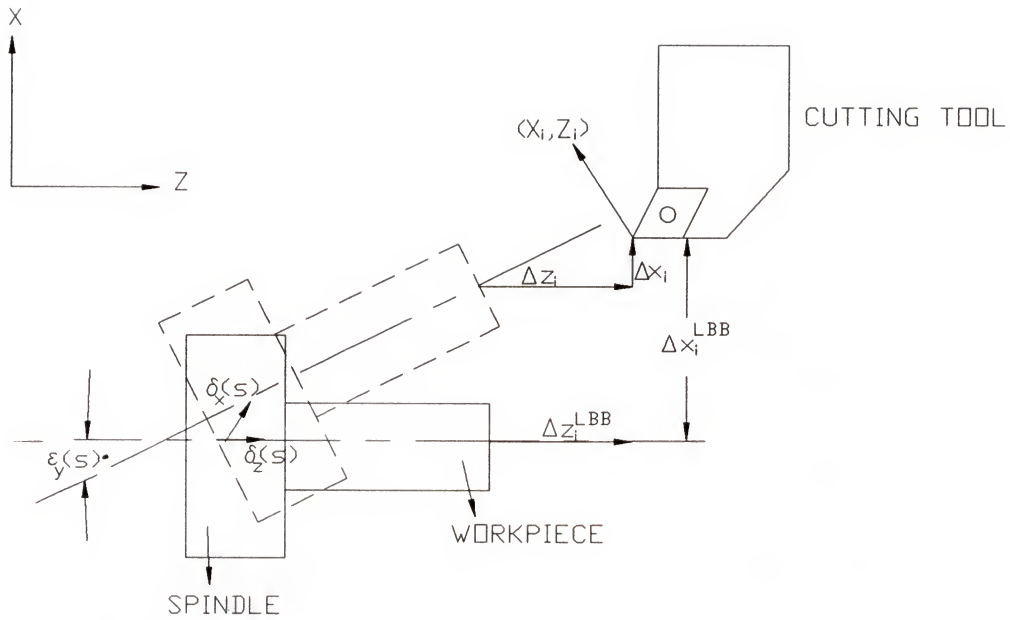


Figure 6-17 Physical intrepretation of effect of spindle drifts on total positional errors.

the network undergoes training and incrementally approximates the mapping between the inputs to ART_A module and ART_B module. The learning time for data collected during each measurement cycle ranges from 3 to 10 minutes on an IBM 486 50MHz computer. Since the learning takes place during the warming cycle (which takes about 20 minutes), the learning time of the network is within limits of the data collection procedure.

Typical examples of positional error map data for the cold state and at thermal equilibrium are plotted in a grid form as shown in figure 6-18 and 6-19. These plots are shown for the forward direction of motion of each axis, and without including spindle drifts. In order to demonstrate the effect of spindle drifts on the positional error map, Figure 6-18 and 6-19 are plotted again with spindle drifts as

shown in Figure 6-20 and 6-21. It can be seen that the spindle drifts cause a significant change in the positional error maps. The difference between Figure 6-18 and 6-20 (both being at cold state), is purely due to the spindle tilt. The differnce between Figure 6-19 and 6-21 (both being at thermal equilibrium) is due to the effect of all the three components of drift errors. Similarly, the effect of axis reversal errors (of each axis) on the total positional error map are plotted in Figures 6-22 and 6-23 for the cold state and at thermal equilibrium respectively.

It should be noted that Figures 6-18 through 6-21 are plotted by normalizing the total positional errors at each grid point to that at the first grid point. While this is adequate to illustrate the effects of thermally induced errors, it cannot be used in this form during the machining of a part on the turning center. This is because the machine coordinates of the first grid point is offset form the machine coordinates of the spindle center as shown in Figure 6-24 in both x and z directions. The x offset causes a diametral error of twice the offset in the machined part. The z offset causes a $\Delta Z\epsilon_y(s)$ error (from equation 6-2) in the x direction. In this research the z motions are measured relative to only one work face of the part and hence the z offset has no effect in the z direction. The x and z offsets at the cold state of TDC1 was determined to be 60 microns and 0.7 microns. Since the effect of the z offset was in the submicron levels, it was neglected. The total positional error map of Figure 6-20 is re-plotted in Figure 6-25 after including the effects of the x offset.

The inputs to ART_A and ART_B networks must be normalized within the range {0, 1}. However, it is not possible to know *a priori*, the range of the thermocouple

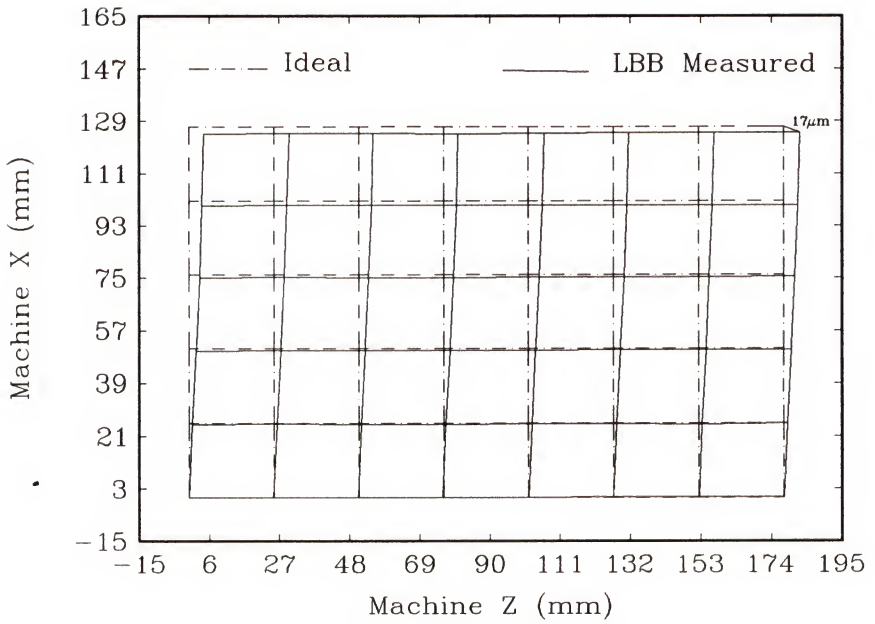


Figure 6-18 Positional error map at cold state of TDC1 without spindle drifts and forward direction.

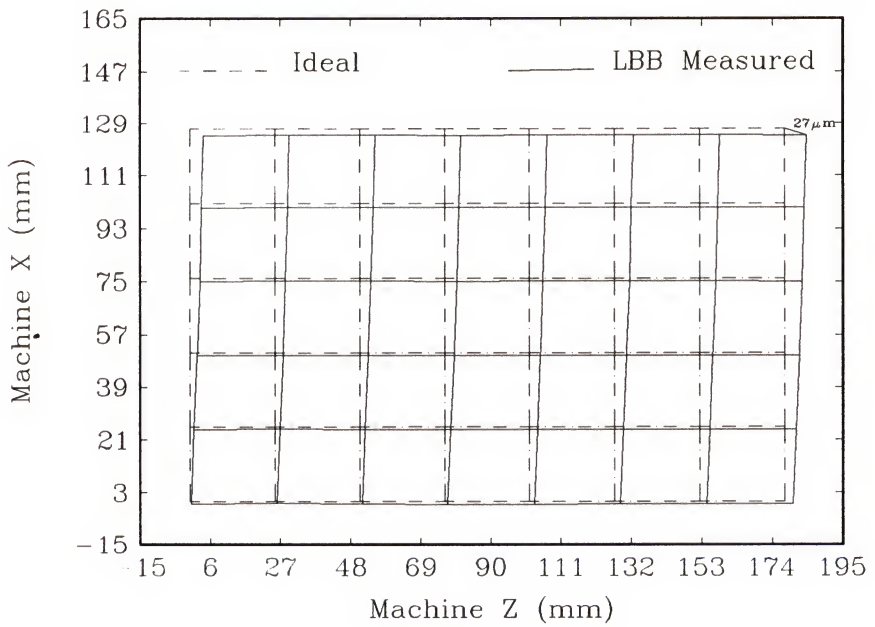


Figure 6-19 Positional error map at thermal equilibrium of TDC1 without spindle drifts and forward direction.

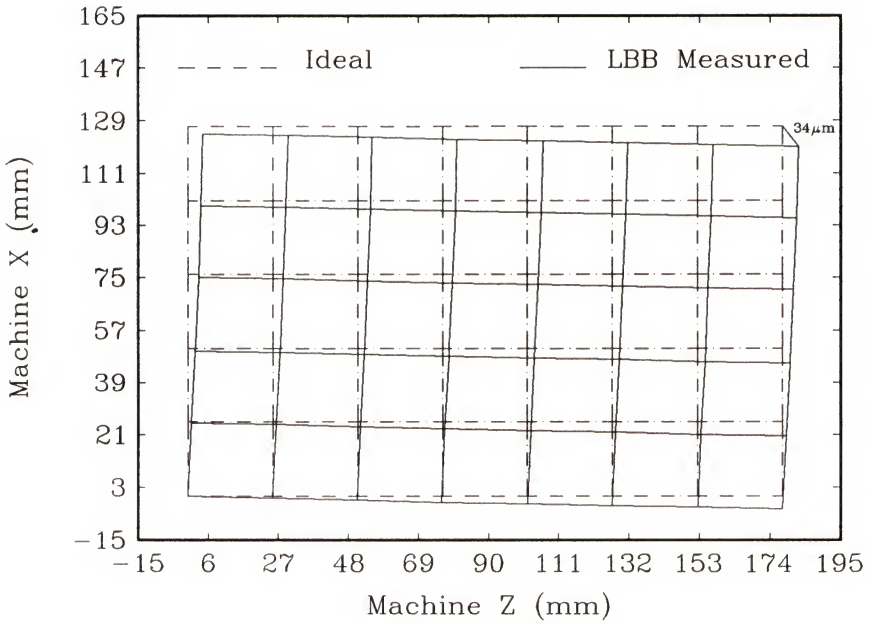


Figure 6-20 Positional error map at cold state of TDC1 with spindle drifts and forward direction.

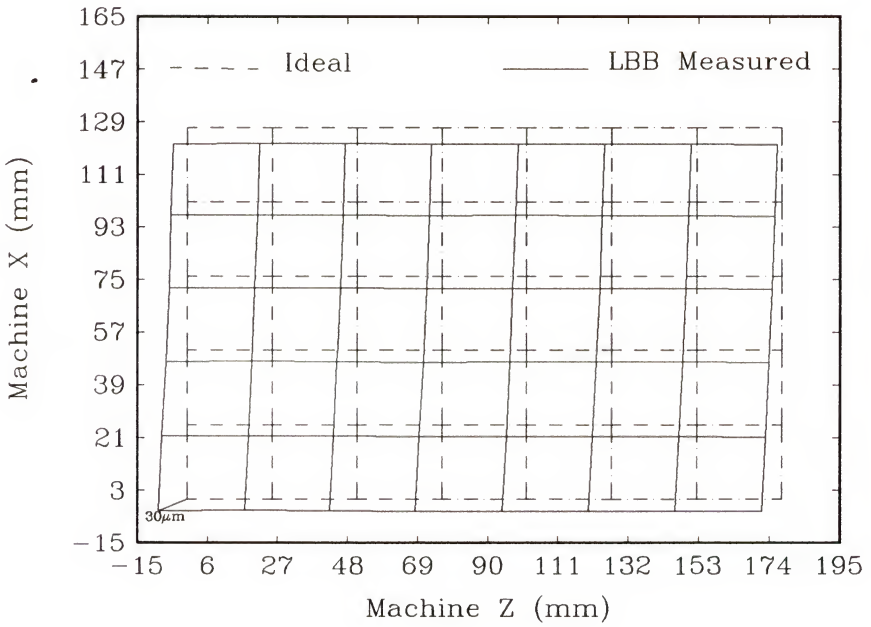


Figure 6-21 Positional error at thermal equilibrium of TDC1 with spindle drifts and without reversal errors.

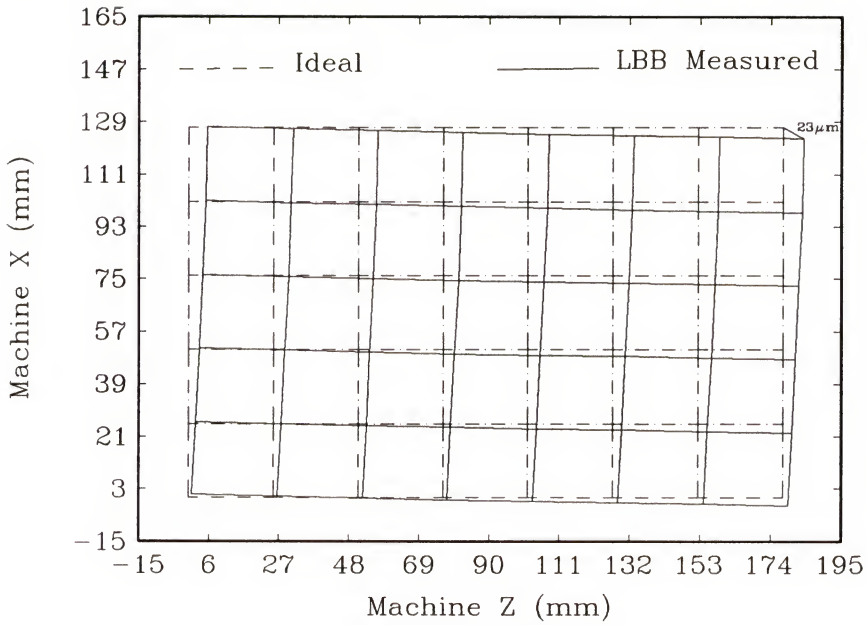


Figure 6-22 Postional error map at cold state of TDC1 with spindle drifts and reverse direction.

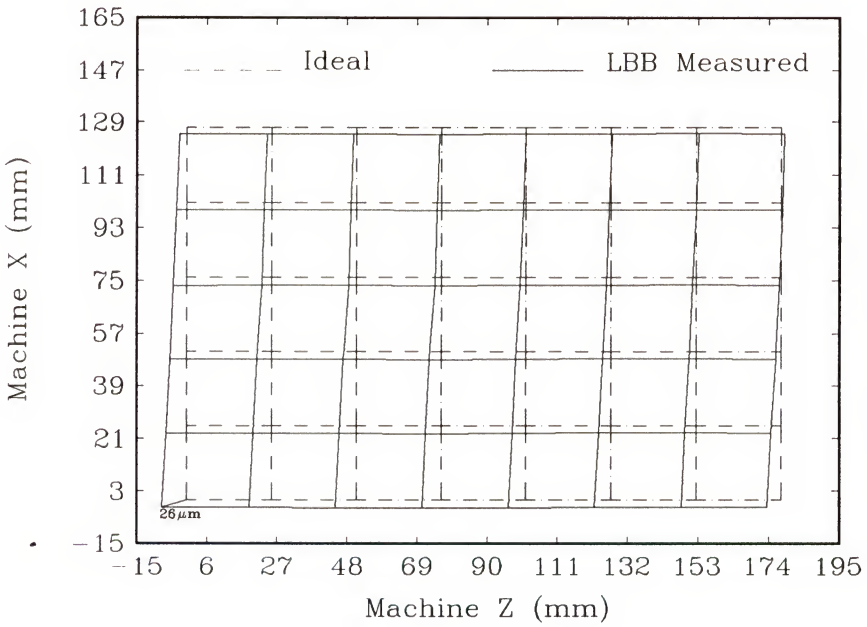


Figure 6-23 Positional error map at thermal equilibrium of TDC1 with spindle drifts and reverse direction.

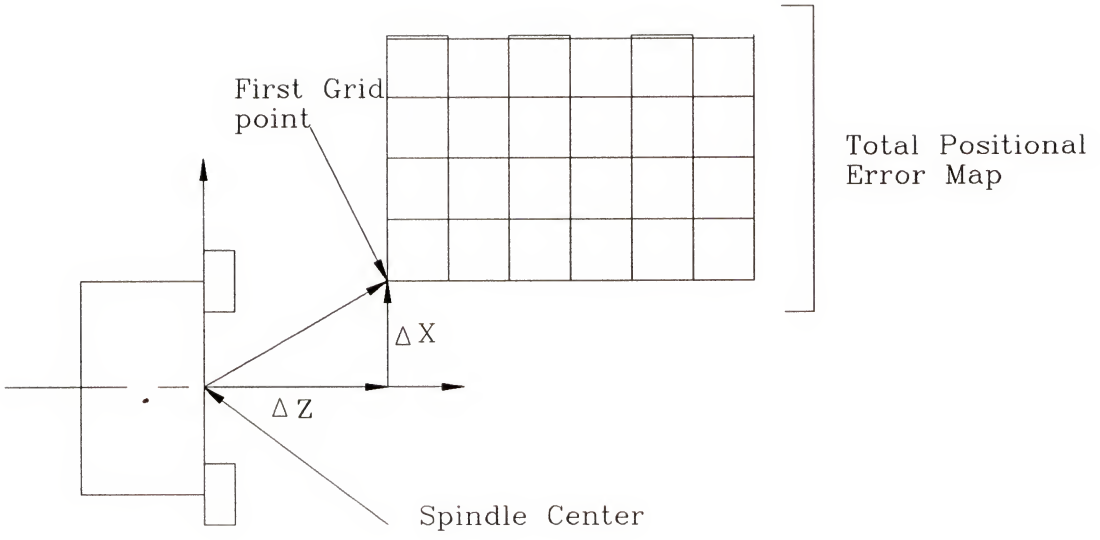


Figure 6-24 Offsets in the x and z direction form spindle center.

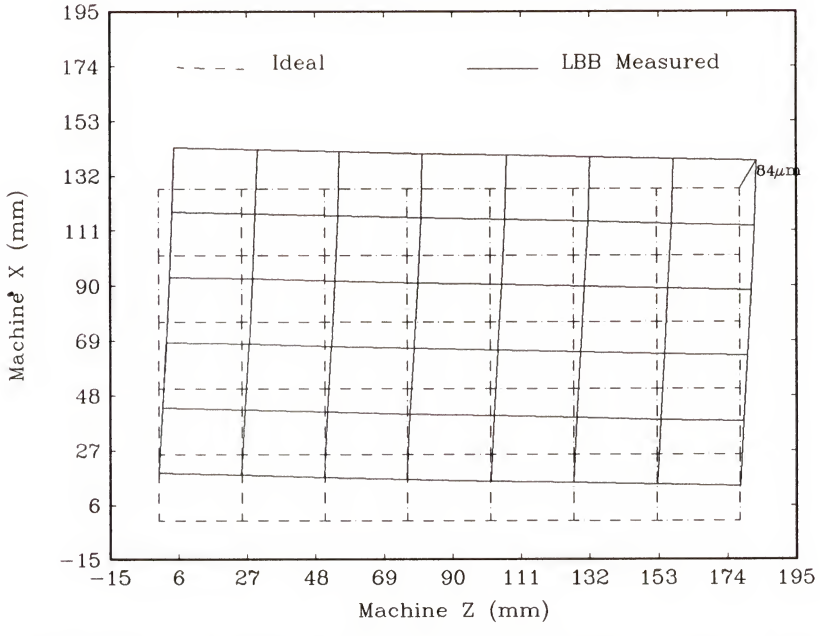


Figure 6-25 Total positional error map that includes spindle drifts, errors due to reversal and offset of first grid point from spindle center.

readings or the range of the errors. There are two solutions to this problem. The first solution evaluates the minimum and the maximum value of each input after each measurement period. If these values are outside the range from the previous measurement period, then data from all previous measurements are re-normalized with respect to the new minimum and maximum values. The Fuzzy ARTMAP is then made to forget all previously learned data (by setting $w^A = w^B = w^{AB} = 1.0$) and relearn the entire data set after re-normalization of all inputs have occurred. The relearning takes place during the next warming period which provides sufficient time for processing the re-normalization scheme. While this solution is portable to any machine tool, the learning is not truly on-line. The second solution sets the ranges of all inputs based on their previous history. This solution results in on-line learning and is feasible in most industries, as sufficient historical knowledge of each machine's characteristics is readily available. Since we had performed the parametric error measurements of the two axis turning center using ASME B5.54 standards earlier, the second solution was adopted.

The neural network was trained on the data collected during TDC1 after including the effects of spindle drifts and the offsets from the spindle center. The training set consisted of 1200 input vectors for TDC1. The baseline vigilance of the ART_A network was set at 0.8; the vigilance parameter of the ART_B network was set to 1.0; By setting the vigilance of ART_B network to 1.0, the system codes the inputs to the F_2^B layer of ART_B network into classes separated by the resolution of the machine (i.e., 1 micron). This is desirable, since the machine cannot compensate for

errors below its resolution. Hence, a setting of 1.0 for the vigilance of ART_B converts the Fuzzy ARTMAP neural network into an adaptive bin classifier where each bin represents a class that is an integral multiple of the machine resolution. The training uses four Fuzzy ARTMAP nets. The 1200 input vectors collected during TDC1 resulted in 629 ART_A classes and 48 ART_B classes when the inputs to ART_B were the z positional errors. Similarly, the network classified the 1200 vectors into 350 ART_A classes and 36 ART_B classes when the inputs to ART_B were the x positional errors. The network classified the 1200 vectors into 631 ART_A and 50 ART_B classes for z (reverse) and 353 ART_A and 34 ART_B classes for x (reverse).

Table 6.1 Test conditions for TDC2.

Trial	AISI#	Feed Rate (mpm)	X Travel (mm)	Z Travel (mm)	Spindle Speed (rpm)	Time (secs)
1	2024	1.950	165	395	1792	1305
2	2011	1.310	43	243	1950	1053
3	7075	1.145	178	378	1202	1789
4	6061	1.770	165	52	1659	961
5	2024	1.580	72	378	1561	1444
6	6061	1.145	135	135	1207	1567
7	2011	1.770	111	391	1141	1605
8	7075	1.950	127	327	1534	1472
9	6061	1.310	63	323	1838	1565
10	2024	1.950	40	300	1945	1066
11	2011	1.145	145	230	1901	1606

In order to test the predicting ability of the neural network, a new thermal duty cycle (TDC2) was formulated as shown in Table 6-1. Five different alloys of aluminum were selected ranging from 2024 (popularly used for screw machine products, truck wheels, etc) to 7075-T6 (aircraft parts). The KENNAMETAL handbook [100] was used to find recommended feed rates and spindle speeds for each alloy.

During each warming period, a material, corresponding feed rate and spindle speed was randomly selected. Range of travel was randomly chosen to lie between 10% to 100% of the entire range of X and Z travel. The spindle speed was randomly chosen between 1000 and 2000 rpm. The warming period was randomly selected to be between 15 and 30 minutes. At each grid point location, the ART_A network is presented with the 14 thermocouple readings, and the nominal coordinates of the tool socket in machine coordinates as inputs. The ART_B network then predicts a corresponding x and z positional error. This prediction is compared with the actual measurement made by the LBB in order to evaluate the predicting ability of the neural network. A general measure of performance of the Fuzzy ARTMAP, R_{eP}, is computed. It is defined as the mean value over the number of measured points, N, of the absolute difference between the positional error P of each axis (P = X or Z) measured by the LBB and the Fuzzy ARTMAP predicted value of the same error:

$$R_{eP} = \frac{\sum_{i=0}^{N} |P_{i,\,LBB} - P_{i,\,ARTMAP}|}{N} \tag{6-4}$$

For TDC2, R_{ex} and R_{ez} were found to be 2 and 3 microns respectively. These results are good considering that the test set is completely different from the training set and the resolution of the machine is 1 micron. The results for some of the test trials are shown in Figures 6-26 through 6-31. The predicting ability of the network could be further improved if the network were trained on a larger training set. However, the limit chosen here gave satisfactory results and is sufficient to illustrate the proposed technique.

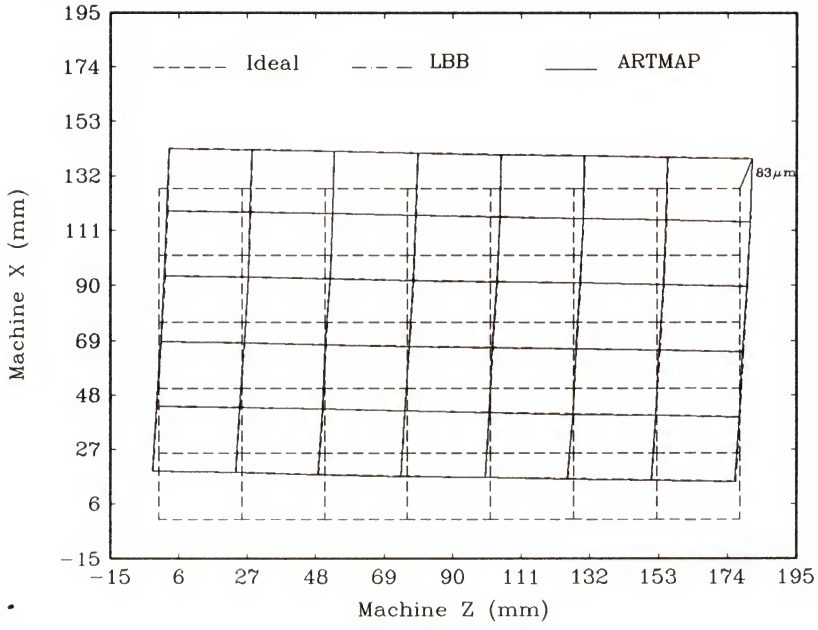


Figure 6-26 Comparison between LBB and ARTMAP for trial #1 of TDC2.

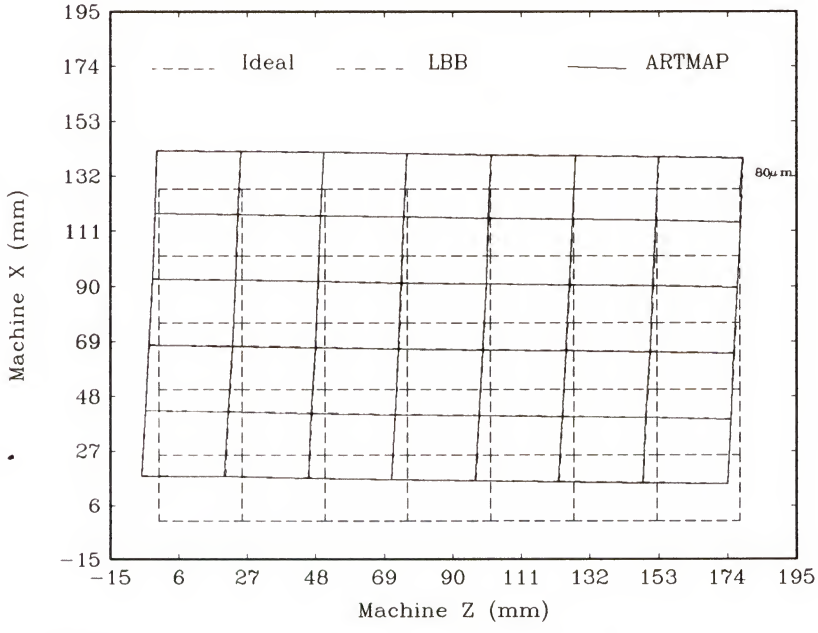


Figure 6-27 Comparison between LBB and ARTMAP for trial#3 of TDC2.

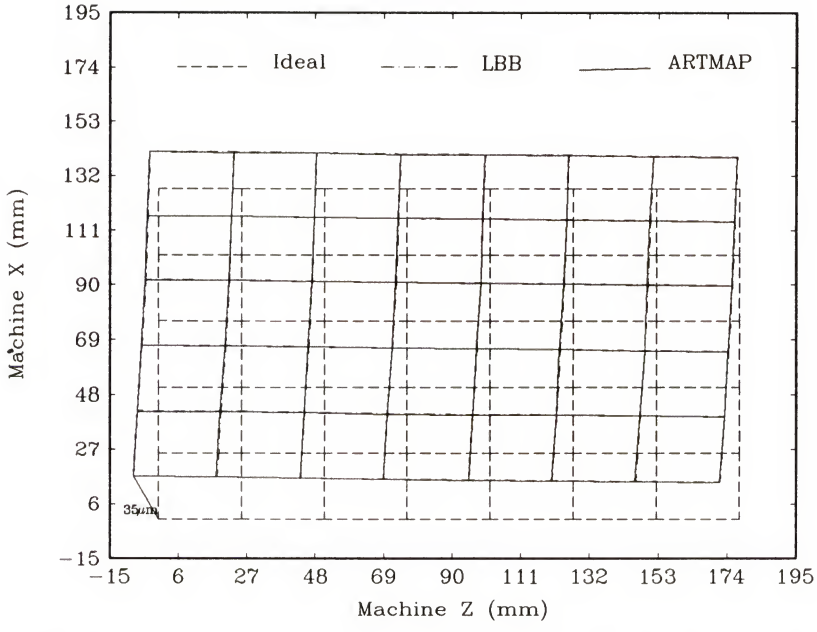


Figure 6-28 Comparison between LBB and ARTMAP for trial#4 of TDC2.

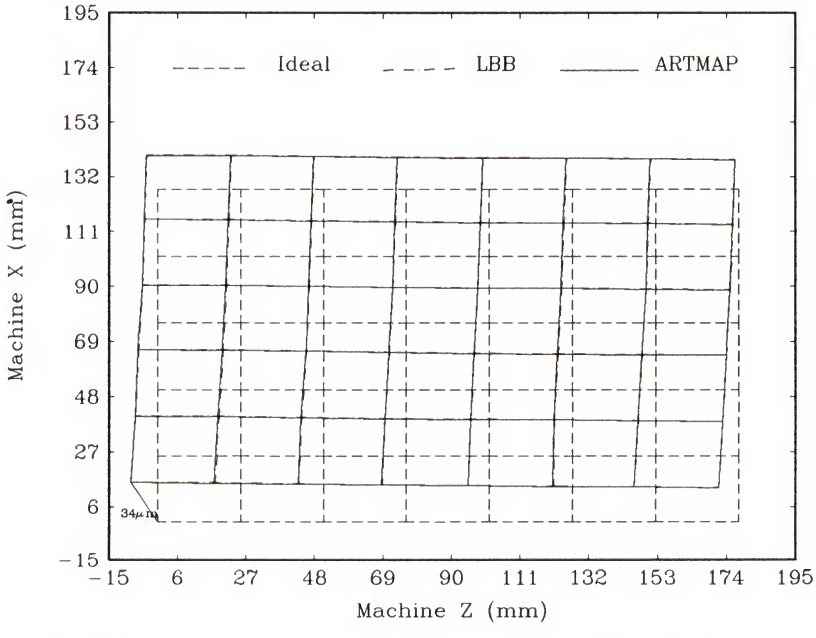


Figure 6-29 Comparison between LBB and ARTMAP of trial#6 of TDC2.

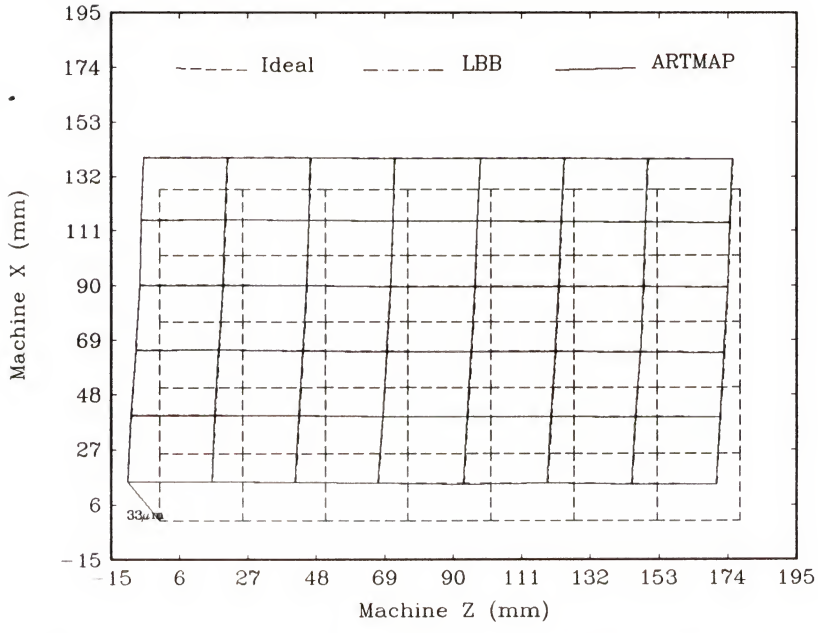


Figure 6-30 Comparison between LBB and ARTMAP for trial #9 of TDC2.

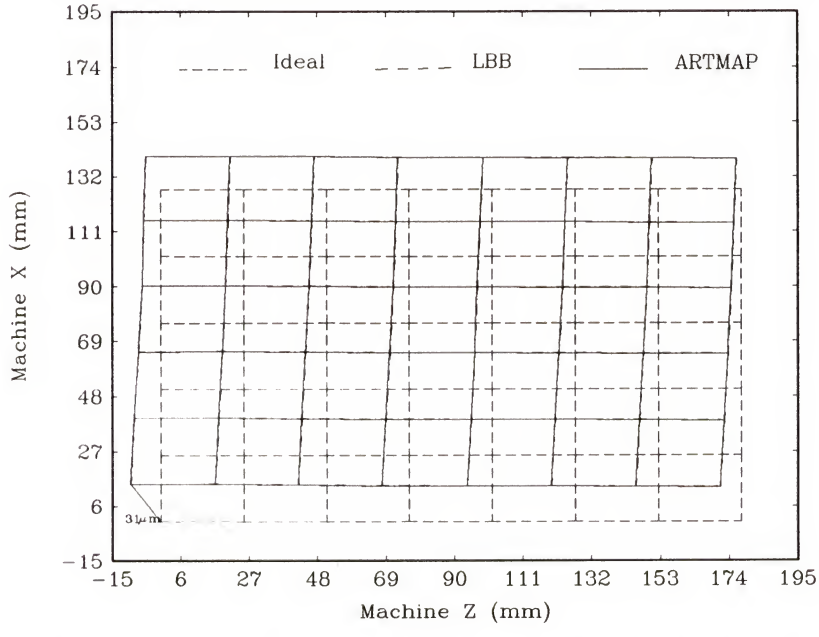


Figure 6-31 Comparison between LBB and ARTMAP for trial #11 of TDC2.

CHAPTER 7
THE ERROR COMPENSATION SYSTEM

In the previous chapter, the error map and spindle drifts of the two axis turning center were collected using the LBB. These errors were learned on-line by the Fuzzy ARTMAP neural network. The predicting ability of the network was tested using a random test cycle. In this chapter, a real-time error compensation system (ECS) developed by Sumanth [80] will be described. This system was used to compensate for machine tool errors on the turning center.

Real-time error compensation is the process of error correction while the machine axes are in motion. There are two methods for real-time error compensation in machine tools. The first method directly modifies the position commands in the controller. In order to implement this approach it is necessary to have an open architecture controller where the control registers are directly accessible. The second method modifies the position feedback signals. Since the MAZAK turning center has a closed architecture, the second approach was adopted by Sumanth in developing the ECS. In this chapter, a brief outline of the hardware and software of the ECS is first presented. This is followed by the details on the the integration of the Fuzzy ARTMAP neural network to the ECS.

Working Principle of the ECS System

The ECS has been implemented on the MAZAK turning center with a MITSUBISHI Meldas CNC controller. The turning center is provided with quadrature incremental encoders on its axes motions to generate feedback signals to the controller. During machining, the controller sends out driving voltages to the axes servo motors based on the desired length and velocity of travel. These servo motors send feedback signals to the controller. The controller keeps track of both the desired and actual positions of the slide and stops the slides when they reach the desired position. The ECS breaks into the feedback loop between the quadrature encoder and the machine tool controllers (MTC) as shown in Figure 7-1. The ECS monitors the encoder signals and then injects or suppresses pulses in the existing encoder waveform. When the ECS adds pulses to the actual quadrature signal, the MTC, which has a quadrature decoder also, counts the injected signals and reaches the position (or the predetermined position count) sooner. By the same reasoning, if some of the axis-encoder pulses are withheld from the MTC, the axis will be kept

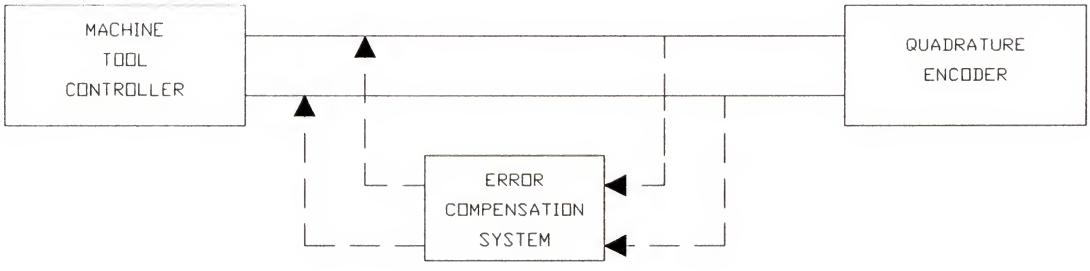


Figure 7-1 Modification in Feedback using the ECS.

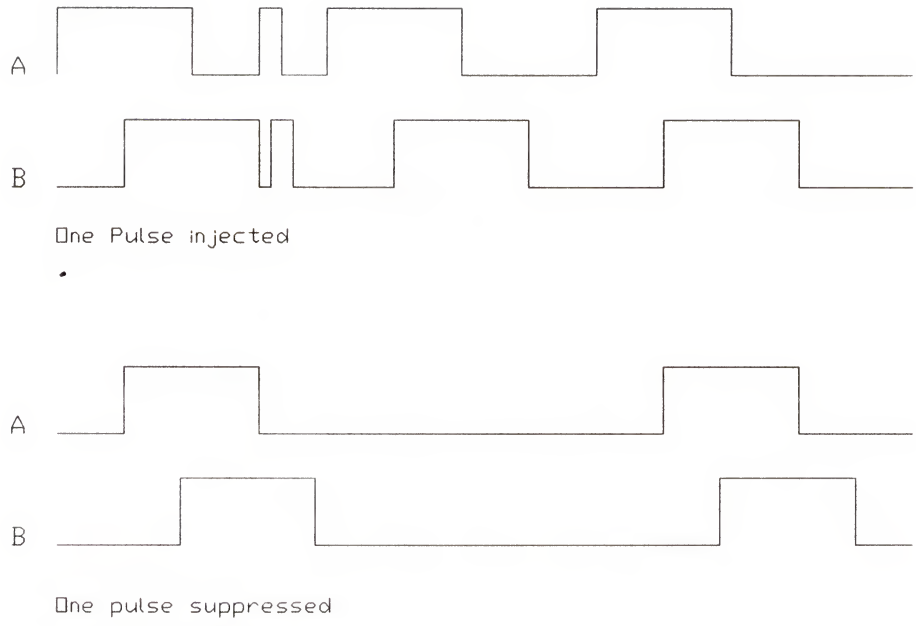


Figure 7-2 Quadrature pulse injection and suppression.

in motion for a longer distance, corresponding to the distance represented by the withheld cycle. The process of injection and suppression of pulses are shown in Figure 7-2. A limitation of the ECS is the resolution of correction. The MTC decodes the quadrature signals with a resolution that is four times finer than the period of the incoming signals (since there are four state changes in every quadrature pulse). However, the ECS implements a correction by injection and suppression of an entire pulse (whole cycle). Hence the compensation resolution is four times coarser than the position resolution.

ECS Hardware

There are five main components in the ECS hardware. They are junction boxes (one for each axis), Quadrature (Q/D) board, a combination of high speed

switches, a quadrature pulse generator, and an IBM 486PC. The junction box was designed and installed by Sumanth [80] at the outlet of the machine encoders in order to tap the quadrature signals. In addition, this box also serves as an interface to the outputs of the compensation system.

The Q/D board is a PC BUS compatible two axis encoder receiver manufactured by Technology 80, Inc. The quadrature train from the junction box is received by the Q/D board. The Q/D board has two 24 bit counters (one for each axis). These counters increment for each state change of the quadrature pulse train. Thus each quadrature pulse increments the counter by four counts. The quadrature pulse train from each axis was calibrated [80] using an oscilloscope and it was found that a complete quadrature pulse corresponds to 2.4 microns of Z axis motion and 1.2 microns of X axis motion. The Q/D board also provides information on the direction of axis movement.

The Q/D board decodes the quadrature pulse train and passes this information in terms of counts to a switching circuit. This circuit consists of high speed CMOS switches. These switches are used to break the encoder-MTC connection and establish the ECS-MTC connection in a manner that is undetected by the machine tool controller. This is possible because the maximum frequency of pulses from the X and Z axis are 200KHz and 100KHz respectively while the high speed switches operate at 0.5MHz. Details of the switching circuit are shown in Figure 7-3. The switches open for a logic zero control input and are closed for a logic one input. These control inputs are provided by the host computer and are

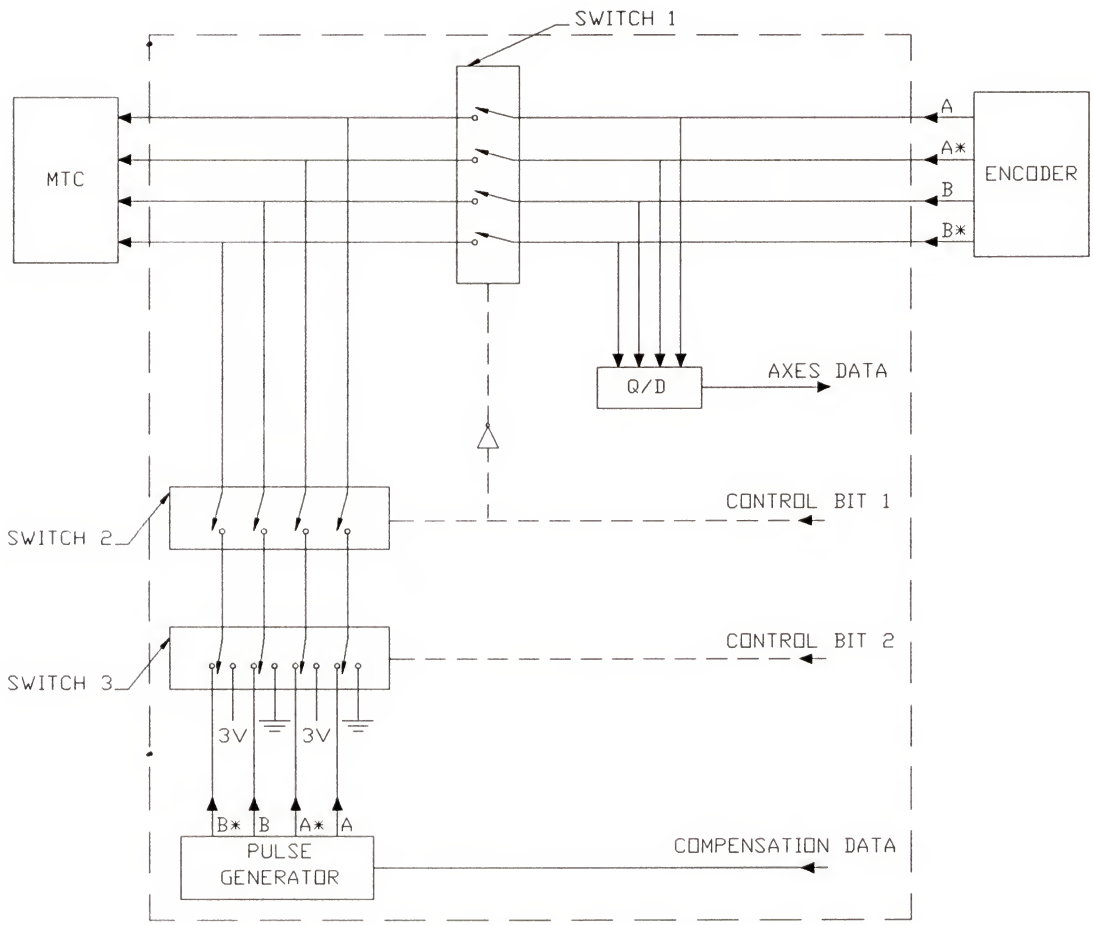


Figure 7-3 Hardware details of the switching circuit [80].

TTL compatible. Switches 1 and 2 are single pole switches while switch 3 is a single pole double throw switch. The characteristics of these switches ensure that during the toggling of a switch, contact with the current connection is broken before connection with the second is established.

The task of pulse injection and suppression is achieved by using a pulse generator. The pulse generator was an EPROM based eight bit microcontroller PIC-16C56 manufactured by Microchip Technology Inc. The pulse generator has a four bit and an eight bit bidirectional port. The four bit port was designated as the output port while the eight bit port was designated as its input port. The pulse generator obtains information about the number of pulses to be generated from the FUZZY ARTMAP neural network via the IBM PC. The frequencies of the pulses to be injected were fixed at 100Khz for Z axis and 200Khz for the X axis. During operation, the microcontroller of the pulse generator continuously polls the most significant bit (MSB) of the input port. When this bit changes to logic one, the information on the number of pulses to be generated is taken as the analog value of the remaining seven bits of the input port. It then waits for the MSB to change to a logic zero state while it polls the least significant bit (LSB) for direction of axis motion. A logic zero denotes positive motion and vice versa. The encoder waveform (for each axis) has two phases A and B with a normal logic state of zero. There are also two phases A^* and B^* (complement of A and B) with a logic state of one (a logic 1 corresponds to 3 volts and a logic zero corresponds to 0 volts). The ECS-MTC connection is first established followed by a break between the pulse generator-

MTC connection. The high speed switch 3 is toggled such that phases A^* and B^* are set to 3 volts and phases A and B are grounded (refer to Figure 7-3). The incoming pulses from the encoder are continuously sampled by the Q/D board and the number of pulses sampled is updated. When this number equals the pulses required to be suppressed, switch 3 is toggled back into its original position and the encoder-MTC connection is reestablished.

ECS Software

The software for the ECS is implemented in Microsoft C++ on an IBM PC. The program continuously samples the axis encoders to keep track of the nominal axes positions and directions. The information is sent to the Fuzzy ARTMAP neural network. The neural network also receives temperature history information from the thermocouples mounted around the machine tool. The network predicts a total positioning error at the cutting tool. When the accumulated positioning error reaches the amount corresponding to one quadrature pulse, the ECS becomes activated.

If the error is positive, a pulse injection takes place. This is achieved by first setting the control bit via software such that switch 1 is opened and switch 2 is closed. This breaks the encoder-MTC connection. Switch 3 is simultaneously connected to the pulse generator. The ECS software is programmed to have a delay proportional to the time taken for a quadrature pulse generation. Upon completion of the injection cycle, switches 1 and 2 are set back to the original states using

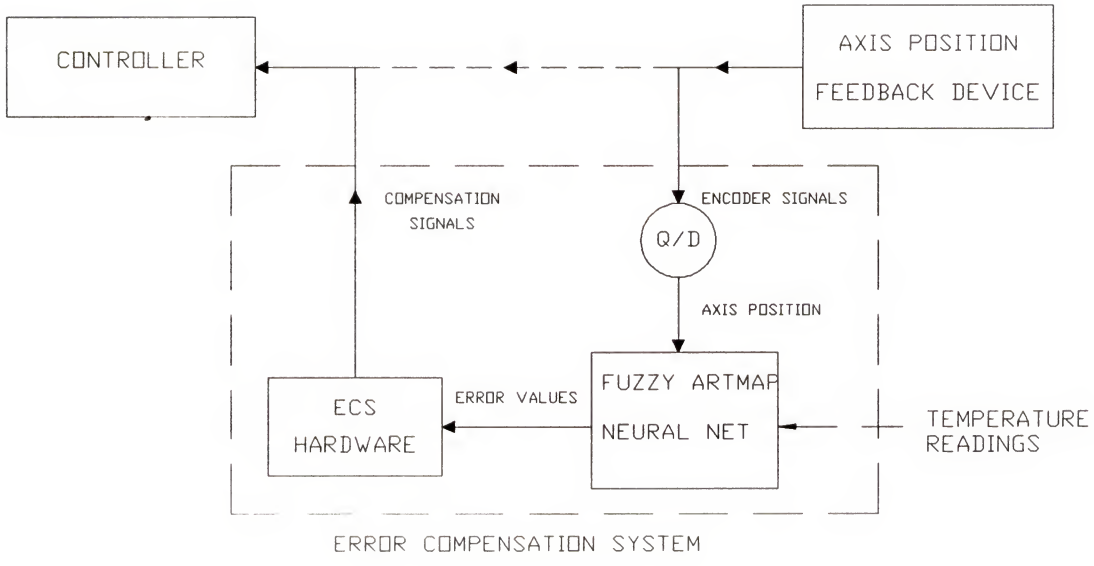
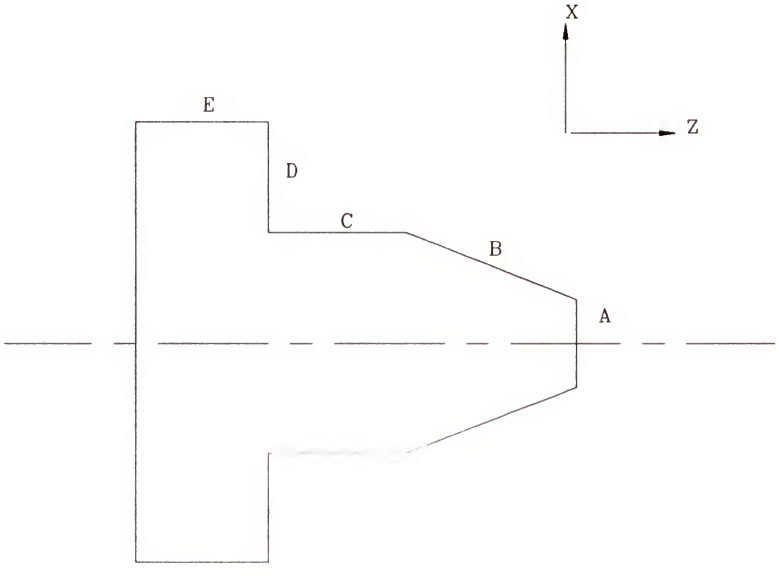


Figure 7-4 Integration of the ECS to the Fuzzy ARTMAP.

control bit 1. If the error is negative, pulses are to be suppressed. The incoming pulse train from the encoder is decoded by the Q/D board to determine if the number of pulses sampled equals the amount of pulses for suppression as provided by the Fuzzy ARTMAP neural network. Switch 3 maintains the normal logic levels of four phases until the desired number of pulses have been sampled. The switches are set back to their original states. The entire ECS system integrated with the Fuzzy ARTMAP is shown in Figure 7-4.

It should be noted that the software waits for the pulse to change state from high to low before it proceeds to inject or suppress pulses. This approach works as well as long as both the axes are in motion. Let us suppose that the Z axis is in motion while the X axis is still. In that case, there is no state change for the X axis. However, as Z moves, some positioning error in the X direction may occur due to

non-straightness of the Z axis or non-parallelism of the Z axis and spindle axis. With the above scheme, the ECS would wait for a state change in the X pulse train that is non-existent and hence will be unable to inject or suppress pulses for the X errors predicted. In order to overcome this problem the encoder pulse train of each axis is monitored for a few milliseconds. If the pulse count for an axis changes by more than 0.5mm, that axis is considered to be in motion. This is a reasonable assumption as the thermally induced errors in the turning center (as measured using the LBB) do not exceed 0.1mm. The ECS software is then provided with the status of motion for each axis in the form of a status file. This file contains two binary parameters (one for each axis). An example of a status file for a part is shown in Figure 7-5. A '1' represents the axis in motion and a '0' represents the axis not in motion. The ECS software waits for a state change of a particular axis to occur (before injecting



SECTION	X	Z
A	1	0
B	1	1
C	0	1
D	1	0
E	0	1

Figure 7-5 Status file compilation for the ECS based on the part shape.

or suppressing pulses) if the binary parameter for that axis is '1'.

The ECS was tested by Sumanth [80] using the LBB. The turning center was programmed to move along a diagonal and the geometric errors were measured with and without compensation. The geometric errors were fitted using multiple regression techniques. The results were encouraging. In this research, thermally induced errors are to be compensated. However, it is not possible to test the ECS for this case using the LBB alone. This is because the effect of spindle drifts cannot be measured simultaneously with the positional errors in a single setting. The only alternative is to evaluate the performance of the ECS by machining parts and compensating for thermally induced errors at various thermal states. These results will be discussed in the next chapter.

CHAPTER 8
PERFORMANCE EVALUATION

The ECS system and Fuzzy ARTMAP neural network will be evaluated for its performance by cutting some workpieces on the turning center. In this chapter the details of the cutting procedure will be presented. This is followed by the results of workpiece measurements and the evaluation of the system in terms of the workpiece dimensions obtained with and without compensation.

Shape of Test Part and Cutting Conditions

A workpiece was specially designed to evaluate the system performance as shown in Figure 8-1. It contains steps of different diameters followed by a taper and a circular arc. The steps help in evaluating the accuracy of X positioning by measuring the diameters, the accuracy of Z positioning by measuring the lengths of the cylindrical surfaces and the squareness between the face of each step and the cylindrical surface. The taper helps in evaluating the accuracy of the two axes in machining a linearly interpolated surface while the circular arc evaluates the ability of the system to accurately machine a circularly interpolated surface.

The material selected for the workpiece was an aluminum alloy of AISI grade 6061. This alloy was chosen for its good machinability. Since workpiece and tool

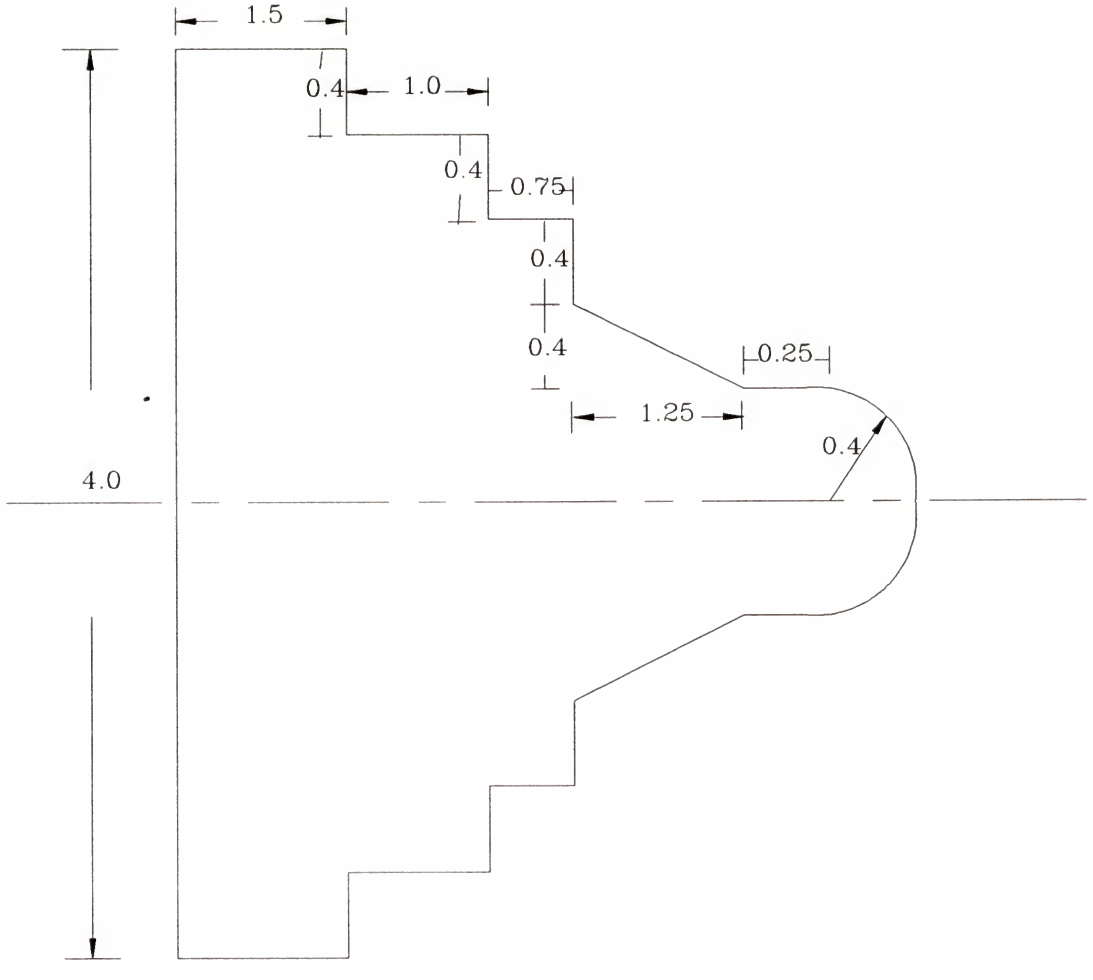


Figure 8-1 Shape of test workpiece.

deflections are not modelled by the neural network, it was necessary to minimize the cutting forces. In order to reduce cutting forces, a $+10^{\circ}$ positive rake angle tool insert (CNMG-UF [100]) was chosen. The insert is also coated with titanium nitride (grade KC730 [100]) to minimize tool wear and edge build up. The insert had a 80° diamond shape (with a one sixty fourth of an inch in nose radius) for ease in both turning and facing operations.

In order to obtain a good surface finish on the workpiece, preliminary cutting trials were performed. The cutting parameters obtained from these trials and subsequently used in the performance evaluation tests are:

Roughing:	Depth of cut	= 1.17mm
	Spindle speed	= 1000 rpm
	Feed rate	= 250 mm/min
Finishing:	Depth of cut	= 0.5 mm
	Surface speed	= 250 m/min
	Feed rate	= 100 mm/min

Cutting Procedure

Three batches of two test parts were machined with and without error compensation at three different thermal states of the machine. The first batch of two parts were machined while the machine was at cold state. Before the machining process begins, the cutting tool was located using the tool eye station. Coolant was used during rough and finish machining of both the parts. The part program (refer

to Appendix C) to machine the test workpiece was written in EIA format on the MAZATROL controller. The machining time for each part was about 12 minutes.

After about 2 hours of warming up, the next batch of two parts were machined with and without error compensation. The machine was then shut down to cool to ambient temperature conditions. It should be noted that the cutting tool was located with the tool eye station at cold state conditions. Similarly, the third batch of two parts were machined the following day after warming the machine continuously for 4 hours.

Measurement Results

Having machined the three batches of test workpieces, the next step was to measure the dimensions of them on a coordinate measuring machine(CMM) in order to evaluate the performance of the ECS system. The parts were measured on an Xcel CMM (manufactured by Brown and Sharpe Company) housed in the Department of Mechanical Engineering at the University of Illinois at Urbana-Champaign. The CMM can be programmed using the language AVAIL (Advanced Validator Interface Language). The accuracy and repeatability of the CMM are 2 and 1 micron respectively. The CMM is equipped to mount Renishaw probes of different tip diameters.

Before the actual measurements can be made on the test workpieces, the CMM requires the tip diameter to be calibrated. This is because the CMM computes the exact location of the probed part after accounting for the offset (i.e.,

radius) of the probe tip. In order to do this, the CMM is provided with a stand containing a reference sphere. The diameter of the reference sphere is precalibrated to be 0.75 inches. The probe tip is used to measure the reference sphere and from that data, the CMM software automatically determines the true diameter of the probe tip. For our measurements, a probe tip of 0.1" was used. The CMM was used to measure the various features of the test workpieces (with and without compensation). The results of these measurements are provided in the following sections.

Error in Diameters

The diameters of each workpiece was measured by best fitting a cylinder on 32 data points taken on each diameter of the best fit cylinder. The CMM software provides the diameter of the best fit cylinder. It also provides the cylindricity of the data points as the zone bounded by two cylinders within which the best fit cylinder lies. The error in diameters $\Delta\phi$ is calculated as the difference between the measured and programmed diameters with and without error compensation. The results are shown in Table 8-1. The dimensions are provided in microinches to be consistent with the units for the dimensions of the test part. The term $\Delta\phi_1^D$ in this table denotes the error in diameter without compensation on the diameter 'D' of the test part while the term $\Delta\phi_2^D$ denotes the same with compensation. This table also provides an improvement ratio R_D due to error compensation as the ratio $\Delta\phi_1^D/\Delta\phi_2^D$. The error in form of the cylinders did not change appreciably before and after

compensation. The average minimum and maximum deviation of the data points from the best fit cylinders (i.e., the cylindricity) was found to be -300μin to 400μin respectively.

Error in Lengths

The lengths of each step was measured with reference to the outer most face of the workpiece. This face was defined as the reference plane to AVAIL by sampling 32 data points around the circumference of the face. Similarly, the face of remaining three steps were defined by planes. AVAIL computes the lengths from the reference plane bycomputing the distance between the centroid of points that define each plane with respect to the reference plane. The error in lengths are calculated by computing the difference between the measured length to the programmed length with and without error compensation. These results are tabulated in Table 8-2. In this table, ΔL_1^l denotes the error in length on length 'l' without compensation while ΔL_2^l denotes the same with compensation. The improvement ratio R_L^l is obtained by computing the ratio $\Delta L_1^l / \Delta L_2^l$.

Error in Taper

The taper on the test workpieces was measured by sampling a total of 40 data points on 5 levels (i.e., height of the cone) with 8 points probed on each level. The 8 points at each level were probed on a circle at that level. The CMM software then provides with the taper angle. The error in taper is calculated as the difference

Table 8-1 Error in Diameters (μin)

T (hrs)	On D = 4.0 in			On D = 3.2 in			On D = 2.4 in			On D = 0.8 in		
	$\Delta\phi_1^{4.0}$	$\Delta\phi_2^{4.0}$	$R_D^{4.0}$	$\Delta\phi_1^{3.2}$	$\Delta\phi_2^{3.2}$	$R_D^{3.2}$	$\Delta\phi_1^{2.4}$	$\Delta\phi_2^{2.4}$	$R_D^{2.4}$	$\Delta\phi_1^{0.8}$	$\Delta\phi_2^{0.8}$	$R_D^{0.8}$
Cold	7700	1300	6.0	8500	1500	5.6	8200	-600	13.7	3200	830	3.85
2	6150	-980	6.3	7300	630	11.6	7100	-760	9.3	6080	740	8.0
4	4600	-700	6.6	1900	640	3.0	5700	430	13.2	4400	680	6.8

Table 8-2 Error in length (μin)

T (hrs)	On L = 1.5 in			On L = 2.5 in			On L = 3.25 in		
	$\Delta L_1^{1.5}$	$\Delta L_2^{1.5}$	$R_L^{1.5}$	$\Delta L_1^{2.5}$	$\Delta L_2^{2.5}$	$R_L^{2.5}$	$\Delta L_1^{3.25}$	$\Delta L_2^{3.25}$	$R_L^{3.25}$
Cold	4000	2000	2.0	5200	900	5.8	4600	500	9.2
2	4500	600	7.5	5700	-1600	3.7	4700	-700	6.7
4	5100	1000	5.1	6100	-700	8.7	5000	-1400	3.6

between the measured taper angle and the programmed taper angle with and without error compensation. These results are tabulated in Table 8-3. In this table, the error in taper without and with compensation are denoted by ΔT_1 and ΔT_2 respectively while the improvement ratio R_T is computed as $\Delta T_1/\Delta T_2$.

Error in Hemisphere Diameters

The hemisphere is measured by using the measurement routine for a sphere available on the CMM. The hemisphere is first located by approximately sampling the north pole of the sphere as well as four points on the equator. The CMM then samples 25 points on the hemisphere with 8 points on three levels. The extra point is the north pole point. The CMM software then provides the diameter of the sphere. The error in hemisphere diameter is calculated by computing the difference between the measured diameter and the programmed diameter with and without compensation. The results are tabulated in Table 8-4. The term ΔD_1 and ΔD_2 denote the measured error in hemisphere diameter with and without compensation respectively. The CMM also provides the sphericity of the hemisphere as the zone bounded by two hemispheres within which the best hemisphere lies. The error in form of the hemispheres did not change appreciably before and after compensation. The minimum and maximum deviation of the data from the best fit hemispheres were -600μin and 400μin respectively. The improvement ratio R_{DH} is calculated as $\Delta D_1/\Delta D_2$.

The error in diameter for the uncompensated part decreases as the machine

Table 8-3 Error in Taper (degrees)

T (hrs)	ΔT_1	ΔT_2	R_T
Cold	0.058	0.0310	1.9
2	0.0197	0.0024	8.3
4	0.0183	-0.0012	15.25

Table 8-4 Error in Hemisphere diameter (μin)

T(hrs)	ΔD_1	ΔD_2	R_{HD}
Cold	3100	820	3.78
2	6100	730	8.35
4	4300	680	6.32

warms up. This is in agreement with the trends seen in the error maps measured using the LBB. The error in lengths for the uncompensated part start increasing as the machine warms up. This is also expected as the total positioning error in the Z direction starts increasing as the machine warms up. The compensation system improves on the diameters and lengths significantly as shown in Table 8-1 and 8-2. The error in taper does not change significantly for this machine with warming as shown in Table 8-3. There was considerable improvement in taper after using the ECS system. These results are encouraging considering that the overall prediction accuracy is 3 microns (about 300μin). In conclusion, these results illustrate that the ECS-Fuzzy ARTMAP neural network combination is capable of error compensation of thermally induced errors in machine tools.

CHAPTER 9
CONCLUSIONS AND FUTURE WORK

Conclusions

Thermally induced errors have been identified as a major contributor to the overall inaccuracy of machine tools. It is possible to compensate for these errors by building a kinematic model of the machine tool. This is achieved by characterizing the parametric errors over a thermal duty cycle. While this method is accurate, it is time consuming and uneconomical for most machine tool users. The main objective of this research was to provide a fast and accurate alternative to the current method.

A direct workspace identification technique was adopted where the total error at the cutting tool is measured directly using a new calibration device called the Laser Ball Bar. This device was used in conjunction with a neural network to rapidly build a model of the machine tool for a thermal duty cycle that simulated machining of large workpieces. The training data for the neural network was collected on a two axis turning center using the LBB. A novel approach was developed to measure thermal drifts and tilts of the spindle using the LBB simultaneously with the error map of the turning center. These errors were transformed into the total error at the cutting tool.

The network incrementally approximated the machine tool error map by

mapping the nominal coordinates of the machine axes and temperatures at certain points on the machine into the total positional errors represented by discrete classes. These classes were separated by the minimum resolution allowable by the machine controller. The prediction ability of the network was then tested on a random thermal duty cycle that simulated machining of small workpieces with short setup times. The overall prediction accuracy of the network was found to be within 3 microns.

The network was then integrated with a PC based error compensation system. The system was interfaced to the encoders of the machine tool as well as thermocouples mounted on the internal heat sources. The system proceeds to inject or suppress quadrature pulses equivalent to the neural network predicted errors into the feedback loop of the machine tool controller. Finally, the performance of the overall system was tested by cutting some test parts under a variety of thermal conditions and substantial improvements in accuracy of the machined parts were achieved using the error compensation system.

Future Work

The following suggestions are provided for future improvements on the system:

1. While the present system models the geometric and thermally induced errors, the network can be trained to learn other sources of errors including static deflections of workpiece and tool and contouring errors. The data for contouring

errors can be measured using the LBB. The workpiece and tool deflections can be measured using force sensors.

2. The Fuzzy ARTMAP is implemented on a PC in software and hence is unable to perform its computations in a parallel processing fashion. A hardware implementation of the neural network using a parallel architecture would substantially improve on its computational speed.

3. The Fuzzy ARTMAP neural network is capable of learning new training data without forgetting its previously acquired knowledge in an on-line fashion. This property makes the network ideally suited for the task of post-process gauging where it can continuously improve its predicting capability based on post-process gauging of machined parts. This property would also help in accounting for wear of the machine components due to constant use.

APPENDIX A
FUZZY ARTMAP NEURAL NETWORK IMPLEMENTATION IN MICROSOFT C

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>

#define npattern 1200          // Total number of training vectors
#define ap npattern            // Maximum number of nodes in ART-A
#define bp npattern/5          // Maximum number of nodes in ART-B
#define alpha 0.001            // ARTMAP parameter
#define adim 13                // Number of features in input to ART_A
#define bdim 1                 // Number of features in input to ART_B
#define pi acos(-1.0)

                // Variables definition for ARTMAP

int ra,rb,ncnf,nodea,nodeb;
```

```
float rhoa,rhob,brhoa;
float *wa[ap],*wb[bp],*tb,*ta;
float a[2*adim],b[2*bdim];
int *wab[ap],*yb,*acnt,*bcnt;

        // Prototype functions of ART_A and ART_B

int fuz_arta(float *wa[],int ra);
int fuz_artb(float *wb[],int rb);

        // FUZZY ART-A NETWORK

int fuz_arta(float *wa[],int ra)
{
    int i,j,count,counter,status,num,nodea;
    float max,vala;
    float sum,sum1,sum2;

        status = 0.0; counter = -1;nodea = 0;

        for(i=0;i<ra+1;i++) {
        *(ta+i) = 0.0;
```

```
*(acnt+i) = 0; }

        while (status == 0) {

      // Calculate the maximally active node in F2

        for(i=0;i<ra+1;i++) {
        sum = sum1 = 0.0;
        for(j=0;j<2*adim;j++) {
        if(a[j] <= *(wa[i]+j)) sum += a[j];
        if(a[j] > *(wa[i]+j)) sum += *(wa[i]+j);
        sum1 += *(wa[i]+j); }
        *(ta+i) = sum/(alpha+sum1); }

         if(counter >= 0) {
         for(j=0;j<counter;j++) { num = *(acnt+j);
         *(ta+num) = -1; }}

        max = 0.0;
        for(i=0;i<ra+1;i++) {
         if(max < *(ta+i)) { max = *(ta+i);
         nodea = i; } }
```

```
            sum = sum2 = 0.0;
            for(j=0;j<2*adim;j++) {
            if(a[j] <= *(wa[nodea]+j)) sum += a[j];
            if(a[j] > *(wa[nodea]+j)) sum += *(wa[nodea]+j);
            sum2 += a[j]; }

             vala = sum/sum2;

                    // Check for Resonance

             if(vala >= rhoa) status = 1;

                        // Check for Reset

             if(vala < rhoa) { status = 0;counter++;
             *(acnt+counter) = nodea; }

             } /* while (status == 0) */

return(nodea);

} // End of fuz_arta()
```

```
// FUZZY ART-B NETWORK

int fuz_artb(float *wb[],int rb)
{
        int i,j,count,counter,status,num,nodeb;
        float max,vala;
        float sum,sum1,sum2;

            status = 0.0; counter = -1;nodeb = 0;
            count = 0;

            for(i=0;i<rb+1;i++) {
            *(tb+i) = 0.0;
            *(bcnt+i) = 0; }

                while (status == 0) {

                /* Calculate the maximally active node in F2 */

                for(i=0;i<rb+1;i++) {
                 sum = sum1 = 0.0;
```

```
for(j=0;j<2*bdim;j++) {
if(b[j] <= *(wb[i]+j)) sum += b[j];
if(b[j] > *(wb[i]+j)) sum += *(wb[i]+j);
sum1 += *(wb[i]+j); }
*(tb+i) = sum/(alpha+sum1); }

if(counter >= 0) {
for(j=0;j<counter;j++) { num = *(bcnt+j);
*(tb+num) = -1; }}

max = 0.0;
for(i=0;i<rb+1;i++) {
if(max < *(tb+i)) { max = *(tb+i);
nodeb = i; } }

sum = sum2 = 0.0;
for(j=0;j<2*bdim;j++) {
if(b[j] <= *(wb[nodeb]+j)) sum += b[j];
if(b[j] > *(wb[nodeb]+j)) sum += *(wb[nodeb]+j);
sum2 += b[j]; }

vala = sum/sum2;
```

```
                    // Check for Resonance

            if(vala >= rhob) status = 1;

                    // Check for Reset

            if(vala < rhob) { status = 0;counter++;
            *(bcnt+counter) = nodeb; }

            } /* while (status == 0) */

            if(vala >= rhob) {
                for(i=0;i<bp;i++) {
                if(i == nodeb) *(yb+i) = 1;
                if(i != nodeb) *(yb+i) = 0; }
                } /* if(val >= rhob) */

    return(nodeb);

} // End of fuz_artb()
```

```
// FUZZY ARTMAP NETWORK

void main()
{
    int i,j,k1,l,status,naind,nval,m,ans,ans1,ans2;
    int mn,vnum,mnodea,mnodeb,pra,prb,ira,irb,cindex,incnt;
    float max,valab,sum,sum1,sum2,bval;
    float bv,valb;
    FILE *pf,*pf1,*fp,*fp1,*fp2,*fp3;
    char fn[20],fn1[20],fn2[20],fn3[20],fn4[20];

    // Static memory allocation for ART_A and ART_B modules

    yb = (int *) malloc(bp*sizeof(int));
    ta = (float *) malloc(ap*sizeof(float));
    tb = (float *) malloc(bp*sizeof(float));
    acnt = (int *) malloc(ap*sizeof(int));
    bcnt = (int *) malloc(bp*sizeof(int));

                    // User Input Session

    printf("Do you want to learn (1 for yes) :");
```

```
scanf("%d",&ans2);

if(ans2 == 1) {

printf("Input name of file to process :");
scanf("%s",fn);

        // Set the vigilance parameters for ART_A and ART_B

printf("Input rhoa :");
scanf("%f",&brhoa);
printf("Input rhob :");
scanf("%f",&rhob);

                // If training for the first time set ra and rb to zero
                // If not, set ra to number of nodes already coded in ART_A
                // rb to number of nodes already coded in ART_B, and ncnt
                // to number of nodes coded in map field

printf("Input ra :");
scanf("%d",&ra);
printf("Input rb :");
```

```
scanf("%d",&rb);
printf("Input ncnt :");
scanf("%d",&ncnt);
if((ra != 0) && (rb != 0)) {
  printf("Input mnodea :");
  scanf("%d",&mnodea);
  printf("Input mnodeb :");
  scanf("%d",&mnodeb);
  printf("Input pra :");
  scanf("%d",&pra);
  printf("Input prb :");
  scanf("%d",&prb); }

printf("Input wa file name:");
scanf("%s",fn1);
printf("Input wb file name:");
scanf("%s",fn2);
printf("Input wab file name:");
scanf("%s",fn3);

 pf = fopen(fn,"r");
```

```
if((ra == 0) & (rb == 0)) {

cindex = 0;ira = irb = -1;

pra = prb = -1;

mnodea = mnodeb = -1;

 fp =.fopen(fn1,"w");

 fp1 = fopen(fn2,"w");

 fp2 = fopen(fn3,"w");

            // Dynamic memory allocation of weights

for( i=0;i<ra+1;i++)

wa[i] = (float *) malloc(2*adim*sizeof(float));

for(i = 0;i<ncnt+1;i++)

wab[i] = (int *) malloc(2*sizeof(int));

for(i = 0;i<rb+1;i++)

wb[i] = (float *) malloc(2*bdim*sizeof(float));

                  // Weights initialization
```

```
    for(i = 0;i<ra+1;i++)
     for(j=0;j<2*adim;j++) *(wa[i]+j) = 1.0;

     for(i = 0;i<rb+1;i++)
     for(j=0;j<2*bdim;j++) *(wb[i]+j) = 1.0;

     for(i = 0;i<ncnt+1;i++)
     for(j = 0;j<2;j++) *(wab[i]+j) = -1;

  }

if((ra != 0) && (rb != 0)) {

    cindex = 1;ira = ra;irb = rb;incnt = ncnt;

     fp = fopen(fn1,"r");
     fp1 = fopen(fn2,"r");
     fp2 = fopen(fn3,"r");

    for( i=0;i<ra+1;i++)
    wa[i] = (float *) malloc(2*adim*sizeof(float));
```

```
    for(i = 0;i<ncnt+1;i++)
    wab[i] = (int *) malloc(2*sizeof(int));

    for(i = 0;i<rb+1;i++)
    wb[i] = (float *) malloc(2*bdim*sizeof(float));

for(i = 0;i<ra+1;i++)
  for(j = 0;j<2*adim;j++)
     fscanf(fp,"%f",(wa[i]+j));

for(i = 0;i<rb+1;i++)
  for(j = 0;j<2*bdim;j++)
     fscanf(fp1,"%f",(wb[i]+j));

for(i = 0;i<ncnt+1;i++)
  for(j = 0;j<2;j++)
     fscanf(fp2,"%d",(wab[i]+j));

 fclose(fp1);fclose(fp2);fclose(fp3);

    } /* if(ra != 0).... */
```

```
for(k1=0;k1<npattern;k1++) {

status = 0;

// Scan data file containing normalized data vectors for ART_A and ART_B

fscanf(pf,"%f %f %f %f %f %f %f %f %f %f %f %f %f
%f",&a[0],&a[1],&a[2],&a[3],&a[4],&a[5],&a[6],&a[7],
&a[8],&a[9],&a[10],&a[11],&a[12],&b[0]);

                    // Set Baseline vigilance for ART-A */

rhoa = brhoa;
valab = 0.0;

                    // Incorporate Complement Coding */

for(i = adim;i<2*adim;i++) {
        a[i] = 1 - a[i-adim]; }

for(i = bdim;i<2*bdim;i++) {
        b[i] = 1 - b[i-bdim];}
```

```
for(i = 0;i < bp;i++) *(yb+i) = 0;

        // Compute nodeb from ART_B module

            nodeb = fuz_artb(wb,rb);

      // Check if reset occurs in ART_A due to map field

      while ((status == 0) && (rhoa <= 1.0)) {

        // Compute nodea from ART_A module

            nodea = fuz_arta(wa,ra);

// Compute the association between ART_A and ART_B in map field

      naind = -1;
      for(i=0;i<ncnt+1;i++){
      if((*wab[i]) == nodea) naind = i;}

      sum = 0.0;sum1 = 0.0;
      for(i = 0;i<bp;i++) {
```

```
if(naind < 0) sum += *(yb+i);
if(naind >= 0) {
if(*(wab[naind]+1) == i) sum += *(yb+i);}
sum1 += *(yb+i);}

valab = sum/sum1;

if(valab < 1.0) { status = 0;
sum = sum1 = 0.0;
for(i = 0;i < 2*adim;i++) {
if(a[i] <= *(wa[nodea]+i)) sum += a[i];
if(a[i] >  *(wa[nodea]+i)) sum += *(wa[nodea]+i);
sum1 += a[i];}
rhoa = 0.005 + (sum/sum1);
} /* if (valab < 1.0) .. */

if(valab == 1.0) status = 1;

} /* while status == 0 .. */
```

// Learning occurs at ART_A and ART_B if no reset occurs from map field

```
if(valab == 1.0) {

  for(j=0;j<2*adim;j++) {
   if(a[j] <= *(wa[nodea]+j)) *(wa[nodea]+j) = a[j];}

  for(j=0;j<2*bdim;j++) {
   if(b[j] <= *(wb[nodeb]+j)) *(wb[nodeb]+j) = b[j];}

        if(mnodea < nodea) mnodea = nodea;
        if(mnodeb < nodeb) mnodeb = nodeb;

        nval = 0;
   // Map Field Learning if Resonance occurs in it

 for(i = 0;i<ncnt+1;i++) {
  if(nodea == *(wab[i])) nval = 1;}

  if(nval == 0) {
*(wab[ncnt]) = nodea;
*(wab[ncnt]+1) = nodeb;
ncnt++; }
```

```
                // Output result for current pattern

        printf("Pattern %d was stored in node %d of ART-A and node %d of
        ART-B\n",k1,nodea,nodeb);

} // if valab == 1.0

        pra = ra;
        prb = rb;

        ra = mnodea + 1;
        rb = mnodeb + 1;

        if(pra == ra) ra += 1;
        if(prb == rb) rb += 1;

        if(pra <= ra) {
        wa[ra] = (float *) malloc(2*adim*sizeof(float));
        for(i =0;i<2*adim;i++) *(wa[ra] + i) = 1.0;}

        if(prb <= rb) {
        wb[rb] = (float *) malloc(2*bdim*sizeof(float));
```

```
        for(i = 0;i<2*bdim;i++) *(wb[rb] + i) = 1.0;}

        if(nval == 0) {
            wab[ncnt] = (int *) malloc(2*sizeof(int));
           for(i = 0;i<2;i++) *(wab[ncnt] + i) = -1;}

    } // for(k = 0;k< npattern

fclose(pf);
                                // Save weights
        if(cindex == 0) {

        for(i = 0;i<ra+1;i++) {
        for(j = 0;j<2*adim;j++) {
        fprintf(fp,"%f ",*(wa[i]+j));}
        fprintf(fp,"\n");}

        for(i = 0;i<rb+1;i++) {
            for(j = 0;j<2*bdim;j++) {
                fprintf(fp1,"%f ",*(wb[i]+j));}
                fprintf(fp1,"\n");}
```

```
            for(i = 0;i<ncnt+1;i++) {
                for(j = 0;j<2;j++) {
                    fprintf(fp2,"%d ",*(wab[i]+j));}
                    fprintf(fp2,"\n");}

} // if(cindex == 0)

            if(cindex == 1) {

                fp = fopen(fn1,"a+");
                fp1 = fopen(fn2,"a+");
                fp2 = fopen(fn3,"a+");

            for(i = ira;i<ra+1;i++) {
             for(j = 0;j<2*adim;j++) {
                fprintf(fp,"%f ",*(wa[i]+j));}
                fprintf(fp,"\n");}

            for(i = irb;i<rb+1;i++) {
             for(j = 0;j<2*bdim;j++) {
                fprintf(fp1,"%f ",*(wb[i]+j));}
                fprintf(fp1,"\n");}
```

```
            for(i = incnt;i<ncnt+1;i++) {
             for(j = 0;j<2;j++) {
                fprintf(fp2,"%d ",*(wab[i]+j));}
                fprintf(fp2,"\n");}

} // if(cindex == 1)

    fclose(fp);fclose(fp1);fclose(fp2);

    printf("ra = %d\n",ra);
    printf("rb = %d\n",rb);
    printf("ncnt = %d\n",ncnt);
    printf("mnodea = %d\n",mnodea);
    printf("mnodeb = %d\n",mnodeb);
    printf("pra = %d\n",pra);
    printf("prb = %d\n",prb);

    } // if(ans2 == 1)

    printf("Do you want to test network learning(1 for yes) :");
    scanf("%d",&ans1);
```

```
if(ans1 == 1) {
                    // Input file containing test data

printf("Input file name to process :");
scanf("%s",fn);
printf("Input filename to store results :");
scanf("%s",fn4);
printf("Is the map for X(1) or Z(0) :");
scanf("%d",&ans);

pf = fopen(fn,"r");
pf1 = fopen(fn4,"w");

for(k1 = 0;k1 < npattern;k1++) {

fscanf(pf,"%f %f %f %f %f %f %f %f %f %f %f %f %f
%f",&a[0],&a[1],&a[2],&a[3],&a[4],
&a[5],&a[6],&a[7],&a[8],&a[9],&a[10],&a[11],&a[12],&b[0]);

for(i = adim;i<2*adim;i++)
a[i] = 1.0 - a[i-adim];
```

```
for(i=0;i<ra+1;i++) {
sum = sum1 = 0.0;
for(j=0;j<2*adim;j++) {
if(a[j] <= *(wa[i]+j)) sum += a[j];
if(a[j]. > *(wa[i]+j)) sum += *(wa[i]+j);
sum1 += *(wa[i]+j); }
*(ta+i) = sum/(alpha+sum1); }

max = 0.0;
for(i=0;i<ra+1;i++) {
if(max < *(ta+i)) { max = *(ta+i);
nodea = i; } }

for(i=0;i<ncnt+1;i++)
if(*(wab[i]) == nodea) nodeb = *(wab[i]+1);

 bval = *(wb[nodeb]);

                // De-normalize outputs and save to file

          a[0] = a[0]*126.999992 -201.789993;
          a[1] = a[1]*177.8 - 905.0;
```

```
            if(ans == 1) {
            b[0] = b[0]*0.023624 -0.023124;
            bval = *(wb[nodeb])*0.023624 - 0.023124;}
            if(ans  == 0) {
            b[0] = b[0]*0.049640 - 0.032780;
            bval = *(wb[nodeb])*0.049640 - 0.032780;}

     fprintf(pf1,"%f %f %f %f\n",a[0],a[1],b[0],bval);
     printf("%f %f %f %f\n",a[0],a[1],b[0],bval);

    } // for(k1 = 0;

     fclose(pf);fclose(pf1);

    }  // if(ans1 == 1)

free(wa);free(wb);free(wab);free(ta);free(tb);free(acnt);free(bcnt);
free(yb);

    }  // end of fuzartmap
```

APPENDIX B
METHOD TO TRANSFORM BALL BAR COORDINATES INTO MACHINE COORDINATES

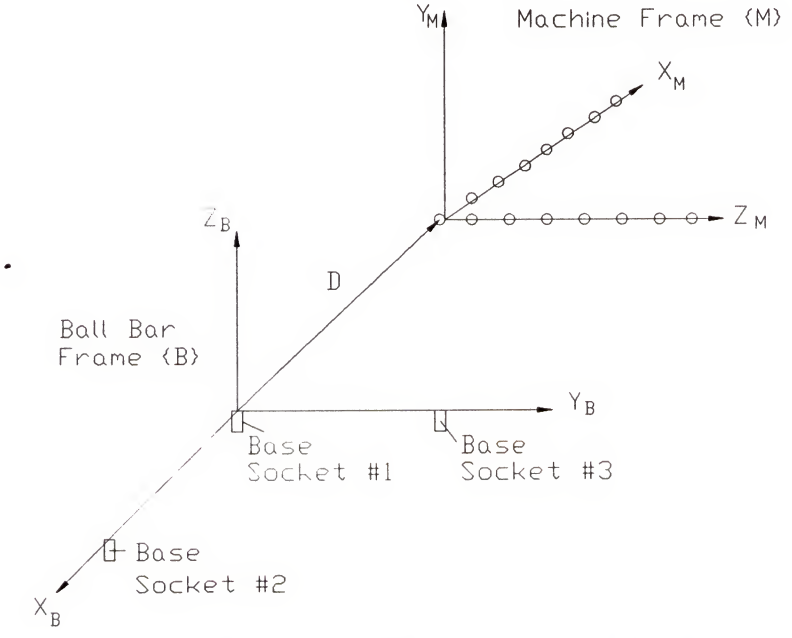


Figure B-1 Coordinate Frame assignments for LBB and machine.

Consider the LBB and machine frames {B} and {M} as shown in Figure A-1. In order to transform the coordinates measured by the LBB in the LBB frame to the machine frame, a reference axis is first defined for the machine frame. In our approach, the reference axis is assumed to be the Z_M axis. The LBB is used to sequentially sample a set of points (x_i, z_i) at regular intervals along the positive Z_M axis. These coordinates are used to best fit a line along this axis. The direction cosines of the best fit line provide the components of the unit vector 'n_Z' in the LBB

coordinates along the Z_M axis. Similarly, a set of points are sampled by moving along the X_M axis. Since the machine axis could have orthogonality errors, the direction cosines of the vector V_X derived from these set of points need not represent a unit vector orthogonal to the Z_M axis. However, n_Z X $V_X/|n_Z$ X $V_X|$ represents a unit vector 'n_Y' in the Y_M direction. By evaluating n_Y X n_Z we arrive at the unit vector 'n_X' along the X_M direction. The unit vectors n_X, n_Y and n_Z form a right handed cartesian coordinate system. It can be seen from [99] that the rotation matrix ${}_BR^M$ matrix between the ball bar frame {B} and machine frame {M} can be expressed as:

$$ {}_B\boldsymbol{R}^{M} = \begin{bmatrix} n_{x1} & n_{x2} & n_{x3} \\ n_{y1} & n_{y2} & n_{y3} \\ n_{z1} & n_{z2} & n_{z3} \end{bmatrix} \tag{B-1} $$

where (n_{X1}, n_{X2}, n_{X3}), (n_{Y1}, n_{Y2}, n_{Y3}) and (n_{Z1}, n_{Z2}, n_{Z3}) are the three components of the unit vector n_X, n_Y and n_Z in ball bar coordinates. The transformation between the two coordinate frames can be expressed as:

$$ {}_B\boldsymbol{T}^{M} = \begin{bmatrix} n_{X1} & n_{X2} & n_{X3} & D_x \\ n_{Y1} & n_{Y2} & n_{Y3} & D_y \\ n_{Z1} & n_{Z2} & n_{Z3} & D_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{B-2} $$

where D_x, D_y and D_z represent the transformation from the origin of frame {B} to the origin of frame {M}. We set the vector to zero when transforming the ball bar coordinates to the machine frame. This is because all the machine errors are normalized with respect to the machine coordinates at the origin of frame {M}.

APPENDIX C
EIA PART PROGRAM FOR TEST WORKPIECE

The CNC part program for the test workpiece consists of two program modules. The first module contains the canned cycle routines for the roughing and finishing operations. It also contains the commands to clamp the spindle speed, run and stop the spindle and also control the flow of coolant. The second module contains the final shape of the part. During the roughing operation, the shape program is used by roughing canned cycle (G71) to determine the number of roughing cuts and the contour to be traced during these cuts. During the finishing operation, the shape program is used by finishing canned cycle (G70) to trace the contour of the part and remove the finishing margins of material to obtain the final part.

MAIN PROGRAM

```
N1 G21 G98;        /* G21 - Metric units: G98 - sets feed rate in mm/min */
N2 G50 S2000 Q0;   /* Clamps spindle speed between 0 and 2000 rpm */
N3 M04 S1000;      /* Run spindle in clockwise direction at 1000 rpm */
N4 M08;            /* Turn coolant on */
/* Roughing canned cycle G71 to rough the stock with 1.17mm roughing
```

depth of cut at a feed rate of 250mm/min and leave 0.5mm in Z and 1mm in X (diametral) as finishing margins */

```
N5 G71 A20 P100 U1000 W500 D1170 F250;
N6 M09;              /* Turn coolant off */
N7 G04 X30000;       /* Delay 30 secs to enable ECS */
N8 MO8;              /* Turn coolant on */
 /* Set surface speed to 250mm/min and turn spindle on */
N9 M04 G96 S250;
N10 G70 A20 P100;          /* Finishing canned cycle G70 */
N11 M09;             /* Turn coolant off */
N12 M05;             /* Turn spindle off */
N13 M02;             /* End of part program */
```

SUB PROGRAM

```
N100 G01 U-127420 F100; /* Sets feed rate at 100 mm/min */
N110 G03 U20320 W-10160 R10160;   /* Circular interpolation */
N120 G01 W-6350;           /* Cylindrical surface */
N130 G01 U20320 W-31750;          /* Taper */
N140 G01 U20320;           /* First step */
N150 G01 W-19050;          /* Cylindrical surface */
N160 G01 U20320;           /* Second step */
N170 G01 W-19050;          /* Cylindrical surface */
```

```
N180 G01 U20320;        /* Third step */
N190 G01 W-38100;       /* Cylindrical surface */
N200 M99;               /* Return to main program */
```

REFERENCES

1 Bowen, G. L., and Duncan, L. S., "Integrated Metrology Systems," SME Technical Paper MS84-759.

2 Hemingray, C. P., "Some Aspects of the Accuracy Evaluation of Machine Tools," 14th MTDR Conference Proceedings, vol. 14, 1973.

3 Chen, J. S., Yuan, J., Ni, J., and Wu, S. M., "Compensation of Non-Rigid Body Kinematic Effect on a Machining Center," Transactions of NAMRI/SME, vol. 20, pp. 325-329, 1992.

4 Blaedal, K. L.,"Error Reduction," Technology of Machine Tools, MTTF, vol. 5 1980.

5 Tlusty, J., "Techniques for Testing Accuracy of NC Machine Tools," Proceedings of the 12th MTDR Conference, pp. 333-345, 1971.

6 Leete, D. L, "Automatic Compensation of Alignment Errors in Machine Tools," International Journal of MTDR, vol. 1, 1961.

7 Wong, G. S. K., and Koenisgberger, F., "Automatic Correction of Alignment Errors in Machine Tools," International Journal of MTDR, vol. 6, 1967.

8 Goodhead, T. C., McGoldrick, P. F., and Crabtree, J. J, "Automatic Detection of and Compensation of Alignment Errors in Machine Tool Slideways," Proceedings of the 18th MTDR Conference, 1977.

9 Bryan, J. B., and Carter, D. L., "Design of a New Error-Corrected Coordinate Measuring Machine," Precision Engineering, vol. 1, 1979.

10 Koliskor, A. S., et al., "Compensating for Automatic Cycle Machining Errors," Machines and Tooling, vol. 41, no. 5, 1971.

11 Schultschick, R., "The Components of Volumetric Accuracy," Annals of the CIRP, vol. 26, 1977.

12 Hocken, R. J., Simpson, J. A., Borchardt, B., Lazar, J., Reeve, C., and Stein, P., "Three Dimensional Metrology," Annals of the CIRP, vol. 26, 1977.

13 Koval, M. I., and Igonin, G. A., "Compensation Analysis of Machining Error Components for a Heavy NC Machine Tool," Machines and Tooling, vol. 50, no. 9, 1979.

14 Portman, V. T., "Error Summation in the Analytical Calculation of Lathe Accuracy," Machines and Tooling, vol. 51, no. 1, 1980.

15 Love, W. J., and Scarr, A. J., "The Determination of the Volumetric Accuracy of Multi-Axes Machines," MTDR Conference Proceedings, vol. 14, 1973.

16 McClure, E. R., "Manufacturing Accuracy Through the Control of Thermal Effects," Ph.D. Dissertation, University of California at Livermore, 1969.

17 Tlusty, J., and Mutch, G. F., "Testing and Evaluating Thermal Deformations of Machine Tools," Proceedings of the 14th MTDR Conference, pp. 285-297, 1973.

18 Okushima, K., Kakino, Y., and Higashimoto, A., "Compensation of Thermal Displacement by Coordinate System Correction," Annals of the CIRP, vol. 24, 1975.

19 Sata, T., Takeuchi, Y., and Okubo, N., "Improvement in the Working Accuracy of a Machining Center by Computer Control Compensation," Proceedings of the 17th MTDR Conference, 1976.

20 Donaldson, R., "A Simple Method for Separating Spindle Error From Test Ball Roundness Error," Annals of the CIRP, vol. 21, 1972.

21 Woyotowitz, M. A., et al., "Tool Path Error Analysis for High Precision Milling With a Magnetic Bearing Spindle," Transactions of the ASME, pp. 129-142, 1989.

22 Wills-Moren, W. J., et al., "Some Aspects of the Design and Development of a Large High Precision CNC Diamond Turning Machine," Annals of the CIRP, vol. 31(1), 1982.

23 Furukawa, et al., "Development of Ultra-Precision Machine Tool Made of Ceramics," Annals of the CIRP, vol. 35(1), 1986.

24 Placek, C., "User Needs Drive CMM Evolution," Quality, Nov. 1990.

25 Takada, K., and Tanabe, I., "Basic Study on Thermal Deformation of Machine Tool Structure Composed of Epoxy Resin Concrete and Cast Iron," Bulletin of Japan Society of Precision Engineering, vol. 21, no. 3, Sept. 1987.

26 Nishiwaki, N., Takeyama, H., and Hori, S., "A Method for Improving the Thermal Behavior of Machine Tools with Heat Pipes," Bulletin of Japan Society of Precision Engineering, vol. 15, no. 4, Dec. 1981.

27 Spur, G., et al., "Thermal Behavior Optimization of Machine Tools," Annals of the CIRP, vol. 37(1), 1988.

28 Weck, M., and Zangs, L., "Computing the Thermal Behavior of Machine Tools Using the Finite Element Method – Possibilities and Limitations," Proceedings of the 16th MTDR Conference, vol. 16, 1975.

29 Venugopal, R., and Barash, M., "Thermal Effects on the Accuracy of Numerically Controlled Machine Tool," Annals of the CIRP, vol. 35(1), 1986.

30 Bryan, J., "International Status of Thermal Research," Annals of the CIRP, vol. 39(2), 1990.

31 Moriwaki, T., "Thermal Deformation and its On-Line Compensation of Hydrostatically Supported Precision Spindle," Annals of the CIRP, vol. 37(1), 1988.

32 Janeczko, J., "Machine Tool Thermal Distortion Compensation," 4th Biennial International Machine Tool Technology Conference, Sept. 1988.

33 Donmez, M. A., "A General Methodology for Machine Tool Accuracy Enhancement: Theory, Application and Implementation," Ph. D. Dissertation, Purdue University, West Lafayette, 1985.

34 Zhang, G., et al., "Error Compensation of Coordinate Measuring Machines," Annals of the CIRP, vol. 34(1), 1985.

35 Teeuwsen, J., Soons, J. A., and Schellekens, P., "A General Method for Error Description of CMM's Using Polynomial Fitting Procedures," Annals of the CIRP, vol. 38(1), 1989.

36 Kurtoglu, M.,"The Accuracy Improvement of Machine Tools," Annals of CIRP, vol. 39, 1990.

37 Anjanappa, M., et al., "Error Correction Methodologies and Control Strategies for Numerical Controlled Machining," Control Methods for Manufacturing Processes, vol. 7, pp. 41-49, 1988.

38 ASME B5.54, "Methods for Performance Evaluation of Computer Numerically Controlled Machining Centers," New York, NY, 1992.

39 Kunzmann, H., et al., "On Testing Coordinate Measuring Machines with Kinematic Reference Standards," Annals of the CIRP, vol. 32(1), 1983.

40 Kunzmann, H., Trapet, E., and Waldele, F., "An Analytical Quadratic Model for the Geometric Errors of a Machine Tool," Journal of Manufacturing Systems, 1986.

41 Neumann, H. J., "Monitoring of Coordinate Measuring Machines with a Test Standard," Computer Integrated Quality System in CIM Systems, IFIP, 1990.

42 Jouy, F., "Theoretical Modelisation and Experimental Identification of the Geometrical Parameters of Coordinate Machines by Measuring a Multi-Directed Bar," Annals of the CIRP, vol. 35(1), pp. 393-396, 1986.

43 Ferriera, P. M., and Liu, C. R., "An Analytical Quadratic Model for the Geometric Error of a Machine Tool," Journal of Manufacturing Systems, 1986.

44 Bryan, J. B., "A Simple Method for Testing Measuring Machines and Machine Tools," Precision Engineering, vol. 4, 1982.

45 Knapp, W., "Circular Test for Three-Coordinate Measuring Machines and Machine Tools," Precision Engineering, vol. 6, no. 2, 1983.

46 Woollett, M., "Rapid Error Correction of Coordinate Measuring Machines for Squareness, Twist, and Scaling Error," Manufacturing Engineering, Oct. 1988.

47 McClure, E. R., "Thermally Induced Errors," Technology of Machine Tools, MTTF, 1980.

48 Hamar, M., "Laser Machine Geometry Measurement and Control," 4th Biennial International Machine Tool Technology Conference, 1988.

49 Chen, J. S., "Real-Time Compensation of Time-Variant Volumetric Error on a Machining Center," Ph.D. Dissertation, University of Michigan, Ann Arbor, 1991.

50 Soons, J. A., Teuws, F. C., and Schellekens, P. H., "Modeling the Errors of Multi-Axis Machines; A General Methodology," Precision Engineering, pp. 5-19, 1992.

51 Hatamura, Y., Nagao, T., Mitsuishi, M., Kato, K., Taguchi, S., Okumura, T., Nakagawa, G., and Sugishita, H., "Development of an Intelligent Machining Center Incorporating Active Compensation for Thermal Distortion," Annals of the CIRP, pp. 549-552, vol. 42(1), 1993.

52 Jedrzejewski, J., and Modrzycki, W., "A New Approach Modeling Thermal Behavior of a Machine Tool Under Service Conditions," Annals of the CIRP, pp.455-

458, vol. 41(1), 1992.

53 Matsumura, T., Obikawa, T., Shirakashi, T., and Usui, E., "An Adaptive Prediction of Machining Accuracy in Turning Operation," Transactions of NAMRI/SME, pp. 305-312, vol. 23, 1994.

54 Mize, C. D, "Design and Implementation of a Laser Ball-Bar Based Measurement Technique for Machine Tool Calibration," Master's Thesis, University of Florida, 1993.

55 Ziegert, J. C., and Mize, C. D., "Laser Ball-Bar: A New Instrument for Machine Tool Metrology," Precision Engineering, (accepted for publication).

56 Chryssolouris, G., Domroese, M., "An Experimental Study of Strategies for Integrating Sensor Information in Machining," Annals of the CIRP, pp. 425-428, vol. 38(1), 1989.

57 Rangwala, S. S., and Dornfeld, D. A., "Learning and Optimization of Machining Operations Using Computing Abilities of Neural Networks," IEEE Transactions on Systems, Man and Cybernetics, vol. 19, no. 2, March/April, pp. 299-314, 1989.

58 Rangwala, S. S., and Dornfeld, D. A., "Sensor Integration Using Neural Networks for Intelligent Tool Condition Monitoring," ASME Journal of Engineering for Industry, vol. 112, pp. 219-228, 1990.

59 Chryssolouris, G., Domroese, M., and Beaulieu, P., "Sensor Synthesis for Control of Manufacturing Processes," ASME Journal of Engineering for Industry, vol. 114, pp. 158-174, 1992.

60 Chryssolouris, G., and Domroese, M., "Sensor Integration for Tool Wear Estimation in Machining," Symposium on Sensors and Controls, ASME WAM, Chicago, IL, pp. 115-123, 1988.

61 Dornfeld, D. A., "Neural Network Sensor Fusion for Tool Condition Monitoring," Annals of the CIRP, vol. 39/1, pp. 101-105, 1990.

62 Dornfeld, D. A., "Unconventional Sensors and Signal Conditioning for Automatic Supervision," AC'90 III CIRP International Conference on Automatic Supervision, Monitoring and Adaptive Control in Manufacturing," Warsaw, Poland, pp. 197-233, 1990.

63 Monostori, L., and Barschdorff, D., "Artificial Neural Networks in Intelligent Manufacturing," Robotics and Computer Integrated Manufacturing, vol. 9, no. 6, pp. 421-437, 1992.

64 Monostori, L., "A Step Towards Intelligent Manufacturing:, Modelling and Monitoring of Manufacturing Processes Through Artificial Neural Networks," Annals of the CIRP, pp. 485-488, 1993.

65 Barschdorff, D., "Case Studies in Adaptive Fault Diagnosis Using Neural Networks," Proceedings of the IMACS Annals on Computing and Applied Mathematics, MIM-s2, Brussels, pp. III.A.1/1-1/6.

66 Teshima, T., Shibasaka, T., Takuma, M., and Yamamoto, A., "Estimation of Cutting Tool Life by Processing Tool Image Data With Neural Network," Annals of the CIRP, pp. 59-62, 1993.

67 Jammu, V. B., and Danai, K., "Unsupervised Neural Network for Tool Breakage Detection in Turning," Annals of the CIRP, pp. 67-70, 1993.

68 Chen, J, S., Yuan, J., Ni, J., and Wu, S. M., "Thermal Error Modeling for Volumetric Error Compensation," ASME Journal of Engineering for Industry, 1993.

69 Kalle, P., and Ziegert, J., "Error Compensation in Machine Tools, a Neural Network Approach," Journal of Intelligent Manufacturing, in press.

70 Rumelhart, D., and McClelland, J., Parallel Distributed Processing: Explorations in the Microstructure of Cognition, vol. 1, MIT Press, 1986.

71 Srinivasa, N., Ziegert, J. C., Kalle, P., and Smith, S., "Neural Network Prediction of Machine Tool Error Maps," Proceedings of the NSF Design and Manufacturing Systems Conference, Atlanta, GA, 1992.

72 Srinivasa, N., Ziegert, J. C, and Smith, S., "Prediction of Thermal Errors in Machine Tools Using Artificial Neural Networks," Proceedings of NSF Design and Manufacturing Systems Conference, Charlotte, NC, pp. 1725-1732, 1993.

73 Carpenter, G.A., Grossberg, S., Markuzon, N., Reynolds, J. H., and Rosen, D. B., "Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps," IEEE Transactions on Neural Networks, vol. 3, no. 5, pp. 698-712, 1992.

74 Lippmann, R. P., "An Introduction to Computing with Neural Networks," IEEE ASSP Magazine, pp. 4 - 22, 1987.

75 Pao, Y. H., Adaptive Pattern Recognition and Neural Networks, Addison-Wesley Publishers, Reading, MA, 1989.

76 Srinivasa, N. and Ziegert, J. C., "Real-Time Learning and Prediction of Thermal

Errors in Machine Tools Using a Fuzzy Logic Based Neural Network," Proceedings of the Symposium on Computer Controlled Machines for Manufacturing, ASME WAM, December, New Orleans, 1993.

77 Smith, S., and Tlusty, J., "Update on High Speed Milling Dynamics," Proceedings of the Symposium on Integrated, Intelligent Manufacturing, ASME WAM, 1987.

78 Smith, S., and Tlusty, J., "Modelling and Simulation of the Milling Process," Proceedings of the Symposium on Sensors and Controls for Manufacturing, ASME WAM, 1988.

79 Tlusty, J., and Smith, S., "Forced Vibration, Chatter, Accuracy in High Speed Milling," Proceedings of the NAMRC, May 1985.

80 Sumanth, D., "Implementation of a Real-Time Error Compensation System on a CNC Turning Center," Master's Thesis, University of Florida, December 1993.

81 Yee, K. W., and Gavin, R. J., "Implementing Fast Probing and Error Compensation on Machine Tools," NISTIR 4447, National Institute of Standards and Technology, Gaithersburg, MD, 1990.

82 Nielsen, R. H., Neurocomputing, Addison-Wesley Publishing Company, 1989.

83 Aarts, E. H. L., and Korst, J. H. M., Simulated Annealing and Boltzmann Machines, New York, Wiley, 1988.

84 Barto, A. G., Sutton, R. S., and Anderson, C. W., "Neuronlike Adaptive Elements that Can Solve Difficult Learning Control Problems," IEEE Transactions on Systems, Man and Cybernetics, vol. SMC-13, pp. 834-846, 1983.

85 Mendel, J. M., and McLaren, R. W., "Reinforcement Learning Control and Pattern Recognition Systems," in Adaptive, Learning and Pattern Recognition Systems: Theory and Applications, Academic Press, New York, pp. 287-318, 1970.

86 Gullapalli, V., "A Stochastic Reinforcement Learning Algorithm for Learning Real-Valued Functions," Neural Networks, vol. 3, pp. 671-692, 1990.

87 Berenji, H. R., "Learning and Tuning Fuzzy Logic Controllers Through Reinforcements," IEEE Transactions on Neural Networks, vol. 3, no. 5, pp. 724-739, September, 1992.

88 Carpenter, G. A., and Grossberg, S., "The ART of Adaptive Pattern Recognition by a Self-Organizing Neural Network," Computer, vol. 21, pp.77-88, 1988.

89 Kohonen, T., "Self-organized formation of topologically correct maps," Biological Cybernetics, vol. 43, pp. 56-69, 1982.

90 Anderson, J., "A Simple Neural Network Generating an Interactive Memory," Mathematical Sciences, vol. 14, pp. 197-220, 1972.

91 Kosko, B., "Adaptive Bidirectional Associative Memories," Applied Optics, vol. 26, pp. 4947-4960, 1972.

92 Srinivasa, N., and Jouaneh, M., "An Invariant Pattern Recognition Machine Using a Modified ART Architecture," IEEE Transactions on Systems, Man and Cybernetics, pp. 1432-1437, September/October, 1993.

93 Carpenter, G. A., Grossberg, S., and Rosen, D. B., "Fuzzy ART: Fast Stable Learning and Categorization of Analog Patterns by an Adaptive Resonance System," Neural Networks, vol. 4, pp. 759-771, 1991.

94 Zadeh, L. A., "Outline of a New Approach to the Analysis of Complex Systems and Decision Processes," IEEE Transactions on Systems, Man and Cybernetics, SMC-3, vol. 1, pp. 28-44, 1973.

95 Mize, C., Ziegert, J. C., Pardue, R., and Zurcher, N., "Spatial Measurement Accuracy Tests of the Laser Ball Bar," A Report for CRADA No. Y-1293-0224 between Martin Marietta Energy Systems and Tetra Precision Inc., August 1994.

96 Zygo, "Axiom 2/20 Laser Measurement System Operation and Reference Manual OMP-0220G," Middlefield, CT, September 1991.

97 Beyer, W. H., CRC Standard Mathematical Tables, (25th ed.), CRC Press Inc., Boca Raton, FL, pp. 285-286, 1981.

98 Hamming , R. W., "Fourier Series and Least Squares," Digital Filters, Prentice-Hall Inc., NJ, pp. 64-65, 1977.

99 Fu, K. S., Gonzalez, R. C., and Lee, C. S. G., Robotics: Control, Sensing, Vision and Intelligence, McGraw Hill Book Company, pp. 20-22, 1987.

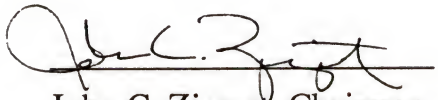
100 The Kennametal Handbook for Machining, Kennametal Inc., Metalworking Systems Division, Raleigh, NC, 1992.

BIOGRAPHICAL SKETCH

Narayan Srinivasa was born on April 22, 1966, in Srivilliputtur, India. He grew up in Madras, India, and then moved to Varanasi, India, to attend the Institute of Technology at Banaras Hindu University, where he received a Bachelor of Technology degree in mechanical engineering in 1988. After working for six months with the Salem Stainless Steel Plant, in Salem, India, he came to the United States in the Spring of 1989 to pursue higher education.

He began his Master of Science program in mechanical engineering at the University of Rhode Island at Kingston, Rhode Island. He completed his Master of Science degree in the Fall of 1990. He then married Anitha and moved to the University of Florida to begin his doctoral program in mechanical engineering in the Fall of 1991. After completing his doctorate, he will be joining the AI group at the Beckman Institute for Advanced Science and Technology, affilated with the University of Illinois at Urbana-Champaign as a post-doctoral research associate.

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

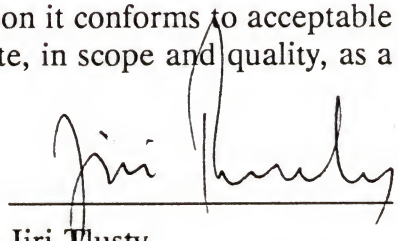


John C. Ziegert, Chairman
Associate Professor of
Mechanical Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

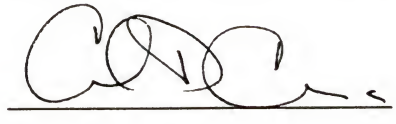
Kevin S. Smith
Associate Professor of
Mechanical Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.



Jiri Tlusty
Graduate Research Professor
of Mechanical Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.



Carl Crane
Associate Professor of
Mechanical Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Baba C. Vemuri
Associate Professor of
Computer and Information
Sciences

This dissertation was submitted to the Graduate Faculty of the College of Engineering and to the Graduate School and was accepted as partial fulfillment of the requirements for the degree of Doctor of Philosophy.

December, 1994

Winfred M. Phillips
Dean, College of Engineering

Karen A. Holbrook
Dean, Graduate School